Nelson-Oppen, Shostak and the Extended Canonizer: A Family Picture with a Newborn

Silvio Ranise, Christophe Ringeissen, and Duc-Khanh Tran

LORIA — INRIA, 615, rue du Jardin Botanique,
BP 101, 54602 Villers-lès-Nancy Cedex France
{ranise, ringeiss, tran}@loria.fr

Abstract. We consider the problem of building satisfiability procedures for unions of disjoint theories. We briefly review the combination schemas proposed by Nelson-Oppen, Shostak, and others. Three inference systems are directly derived from the properties satisfied by the theories being combined and known results from the literature are obtained in a uniform and abstract way. This rational reconstruction is the starting point for further investigations. We introduce the concept of extended canonizer and derive a modularity result for a new class of theories (larger than Shostak and smaller than Nelson-Oppen theories) which is closed under disjoint union. This is in contrast with the lack of modularity of Shostak theories. We also explain how to implement extended canonizers by using the basic building blocks used in Shostak schema or by means of rewriting techniques.

1 Introduction

There is an obvious need of using decision procedures in deduction systems and constraint programming environments since their use allows us to reason on a specific computation domain (or a class of computation domains), to improve efficiency and reduce user-interaction. In almost all applications, the computation domain is an amalgamation of domains or a union (combination) of theories whose domains are axiomatized by formulae. For example, program verification usually assumes a union of theories axiomatizing classical data-structures such as lists, arrays, and arithmetics. To tackle this kind of problems, an appealing approach is to proceed in a modular way, by combining decision procedures available for component theories. This line of research was started in the early 80's by two combination schemas independently presented by Nelson-Oppen [19] and Shostak [24] for unions of theories with disjoint signatures. Each schema makes different assumptions on the properties the theories to be combined should satisfy. The former requires the theories to have a satisfiability procedures and to be such that a satisfiable formula in a component theory T is also satisfiable in an infinite model of T (*stable-infiniteness*). The latter assumes the theories admit procedures for reducing terms to canonical form (*canonizers*) and algorithms for solving equations (*solvers*). A *NO* theory admits a satisfiability procedure and is stably-infinite while a canonizer and a solver are defined for a *SH* theory.

Z. Liu and K. Araki (Eds.): ICTAC 2004, LNCS 3407, pp. 372–386, 2005.

Recently, a series of papers [5, 22, 3, 14, 13, 17, 23, 4, 18] have clarified the subtle issues of combining *SH* theories by studying their relationships with *NO* theories. Unfortunately, these papers lack uniformity and non-experts may be confused. For example, some works [5, 22, 3, 23] use pseudo-code to describe the combination algorithms while others [13, 17, 4, 18] adopt a more abstract (rule-based) presentation. There are advantages (and disadvantages) in both approaches: the pseudo-code offers a better starting point for implementation while inference systems make correctness proofs easier. The **first contribution** of this paper is to provide a synthesis of Nelson-Oppen and Shostak approaches to disjoint combination by using a rule-based approach in which many recent results are recast and proved correct in a uniform, rigorous, and simple way.

Our rational reconstruction proceeds as follows. First, we recall that *SH* theories are contained in the class of (convex) *NO* theories (Section 2.1). According to this abstract classification, three possible scenarios are to be considered when combining two theories: (a) both are *NO* theories (Section 3.1), (b) both are *SH* theories (Section 3.2), and (c) one is a *SH* and the other is a *NO* theory (Section 3.3). We formalize the combination schema for each scenario as an inference system. The applicability conditions of the inference rules are derived from the properties of the theories being combined. Along the lines of [13, 18, 4], the combination schema for (b) is obtained as a refinement of that for (a). The inference system formalizing the combination schema for (c), already considered in [3], is obtained by modularly reusing those for (a) and (b) in a natural and straightforward way. As a final remark, we mention the possibility of refining the abstract inference systems presented here with strategies as done in [4], so to get a more fine-grained rule-based implementation which mimics a Shostak procedure as described in [23]. We do not do this here, since we are interested in modularity rather than efficiency.

Our synthesis of combination schemas serves two purposes. First, although the results are not new, we believe that presenting them in a uniform framework could provide a valuable reference for people interested in combination problems, especially for non-experts of the field. Second, it can serve as the starting point for further investigations. As an example, a problem of greatest importance when combining *SH* theories is the lack of modularity for solvers [17]: no general method exists to produce a solver for the union of *SH* theories from the solvers of the component theories. This lack of modularity together with the observation that the theory of equality (ubiquitous in virtually any application where combinations of decision procedures are needed) is not a *SH* theory seem to suggest that any *ad hoc* combination schema for scenario (c) constitute a reasonable trade-off between efficiency and generality: solvers and canonizers for *SH* theories efficiently derive new equalities and cooperate in a Nelson-Oppen way. This solution (adopted, for example, in ICS [11]) can be easily specified in the framework proposed in this paper. In fact, the schema of Section 3.1 can be applied to construct a satisfiability procedure for the union of many *NO* theories which can then be used as the component *NO* theory in a simple generalization of the schema in Section 3.3 to accommodate several solvers and canonizers.

However, this solution leaves open the question about the existence of a suitable concept that would allow us to obtain a modularity result and retain some of the efficiency of the canonizers and solvers. By investigating this question in our framework, we propose the concept of *extended canonizer* which constitutes the **second contribution** of our paper. Intuitively, an extended canonizer allows us to canonize terms with respect to a given theory T and a given T-satisfiable set of equations Γ, so that the uniform word problem for T, i.e. $T \models \Gamma \Rightarrow s = t$, reduces to the problem of checking the identity $ecan(\Gamma)(s) = ecan(\Gamma)(t)$, where $ecan(\Gamma)(s)$ and $ecan(\Gamma)(t)$ are the "extended canonical forms" of s and t, respectively (Section 4.1). A similar concept was introduced in [22] for the theory of equality and its combination with one Shostak theory is also described by a rigorous version of Shostak schema. In [23], such a schema is generalized to consider the combination of the theory of equality with an arbitrary number of *SH* theories by an interesting generalization of Shostak schema requiring only the construction of a canonizer for the union of the theories and invoking the solvers for the constituent theories. The main difference with our work is that the concept of extended canonizer introduced in this paper is *modular*, i.e. there exists a procedure that, given two extended canonizers for two component theories, yields an extended canonizer for their union (Section 4.3). Another interesting feature of extended canonizers is that they can be *efficiently built* by reusing a wealth of existing techniques such as canonizers and solvers for *SH* theories and rewriting techniques (as advocated in [15, 2, 1]) for theories which do not admit a solver (Section 4.2). To summarize, the concept of extended canonizer offers an interesting trade-off between modularity and the possibility to reuse disparate techniques to solve the uniform word problem under a common interface. As a final remark, we notice that our definition of extended canonizer is orthogonal to the line of research (advocated in [17]) which suggests that modular solvers may exist in modified settings such as multi-sorted logic.

Structure of the Paper. Section 2 introduces basic concepts of first-order logic and combination of theories. Section 3 presents a rational reconstruction of combination methods. Section 4 defines the concept of extended canonizer and shows how it can be built out of canonizers and solvers, or rewriting based procedures; the modularity of extended canonizers is also studied. Section 5 presents some conclusions and discusses the future work. All proofs omitted in this version of the paper can be found in [21].

2 Preliminaries

We assume the usual first-order syntactic notions of signature, term, position, and substitution, as defined, e.g., in [7].

Let Σ be a first-order signature containing only function symbols with their arity and $\mathcal{X}$ a set of variables. A 0-ary function symbol is called a *constant*. A *Σ-term* is a first-order term built out of the symbols in Σ and the variables in $\mathcal{X}$. We use the standard notion of substitution. We write substitution applications in postfix notation, e.g. $t\sigma$ for a term t and a substitution σ. The set of variables

occurring in a term t is denoted by $Var(t)$. If l and r are two Σ-terms, then $l = r$ is a Σ-*equality* and $\neg(l = r)$ (also written as $l \neq r$) is a Σ-*disequality*. A Σ-literal is either a Σ-equality or a Σ-disequality. A Σ-*formula* is built in the usual way out of the universal and existential quantifiers, Boolean connectives, and symbols in Σ. If φ is a formula, then $Var(\varphi)$ denotes the set of free variables in φ. We call a formula *ground* if it has no variable, and a *sentence* if it has no free variables. Substitution applications are extended to arbitrary first-order formulas, and are written in postfix notation, e.g. $\varphi\sigma$ for a formula φ and a substitution σ.

We also assume the usual first-order notions of interpretation, satisfiability, validity, logical consequence, and theory, as given, e.g., in [10]. A *first-order theory* is a set of first-order sentences. A Σ-*theory* is a theory all of whose sentences have signature Σ. All the theories we consider are first-order theories *with equality*, which means that the equality symbol $=$ is always interpreted as the identity relation. The theory of equality is denoted by $\mathcal{E}$. A Σ-structure $\mathcal{A}$ is a model of a Σ-theory T if $\mathcal{A}$ satisfies every sentence in T. A Σ-formula is *satisfiable in* T if it is satisfiable in a model of T. Two Σ-formulas φ and ψ are *equisatisfiable in* T if for every model $\mathcal{A}$ of T, φ is satisfiable in $\mathcal{A}$ iff ψ is satisfiable in $\mathcal{A}$. The *satisfiability problem* for a theory T amounts to establishing whether any given finite quantifier-free conjunction of literals (or equivalently, any given finite set of literals) is T-satisfiable or not. A *satisfiability procedure* for T is any algorithm that solves the satisfiability problem for T.[1] Note that we can use free constants instead of variables to equivalently redefine the satisfiability problem for T as the problem of establishing the consistency of $T \cup S$ for a finite set S of ground literals. The *uniform word problem* for a theory T amounts to establishing whether $T \models \Gamma \Rightarrow e$, where Γ is a conjunction of Σ-equalities, e is a Σ-equality, and all the variables in $\Gamma \Rightarrow e$ are (implicitly) universally quantified.

Given an inference system R composed of inference rules (written as $P \vdash C$), the binary relation $\vdash_{\mathsf{R}}$ is defined on formulas as follows: $\Phi \vdash_{\mathsf{R}} \Phi'$ if Φ' can be derived from Φ by applying a rule in R. The reflexive and transitive closure of $\vdash_{\mathsf{R}}$, denoted by $\vdash^*_{\mathsf{R}}$, is called the *derivation relation* of R. Also, a *derivation* in R is a sequence $\Phi \vdash_{\mathsf{R}} \Phi' \vdash_{\mathsf{R}} \Phi'' \vdash_{\mathsf{R}} \cdots$. A formula Φ is in *normal form w.r.t.* $\vdash_{\mathsf{R}}$ if there is no derivation in R starting from Φ. The relation $\vdash^*_{\mathsf{R}}$ is *terminating* if there is no infinite derivation. We will write the *configuration* Γ, Δ to denote a formula $\Gamma \wedge \Delta$, where Γ is a conjunction of equalities and Δ is a conjunction of disequalities.

2.1 Combination of Theories

In the sequel, let Σ_1 and Σ_2 be two disjoint signatures (i.e. $\Sigma_1 \cap \Sigma_2 = \emptyset$) and T_i be a Σ_i-theory for $i = 1, 2$. A $\Sigma_1 \cup \Sigma_2$-term t is an i-*term* if it is a variable or it has the form $f(t_1, ..., t_n)$, where f is in Σ_i (for $i = 1, 2$ and $n \geq 0$). Notice that

[1] The satisfiability of any quantifier-free formula can be reduced to the satisfiability of sets of literals by converting to disjunctive normal form and then splitting on disjunctions, e.g., checking whether $S_1 \vee S_2$ (where S_1 and S_2 are conjunction of literals) is T-satisfiable reduces to checking the T-satisfiability of both S_1 and S_2.

a variable is both a 1-term and a 2-term. A non-variable subterm s of an i-term is *alien* if s is a j-term, and all superterms of s are i-terms, where $i, j \in \{1, 2\}$ and $i \neq j$. An i-term is *i-pure* if it does not contain alien subterms. A literal is i-pure if it contains only i-pure terms. A formula is said to be *pure* if there exists $i \in \{1, 2\}$ such that every term occurring in the formula is i-pure. We will write the *configuration* $\Phi_1; \Phi_2$ to denote a formula $\Phi_1 \wedge \Phi_2$, where Φ_i is a conjunction of i-pure literals ($i = 1, 2$).

In this paper, we shall consider the problem of solving the satisfiability problem for $T_1 \cup T_2$ (i.e. the problem of checking the $T_1 \cup T_2$-satisfiability of conjunctions of $\Sigma_1 \cup \Sigma_2$-literals) by using the satisfiability procedures for T_1 and T_2. For certain theories, more basic algorithms exist which can be used to build satisfiability procedures, e.g. canonizers and solvers for the class of Shostak theories (see below for a formal definition). When such algorithms exist for either T_1, T_2, or both, we are interested in using them to solve the satisfiability problem for $T_1 \cup T_2$. In order to know which basic algorithms are available for T_1 and T_2 and what are the assumptions on T_1 and T_2, the following notions and results are useful.

Definition 1. [20] A conjunction Γ of Σ-literals is convex in a Σ-theory T iff for any disjunction $\bigvee_{i=1}^{n} x_i = y_i$ (where x_i, y_i are variables and $i = 1, ..., n$) we have that $T \cup \Gamma \models \bigvee_{i=1}^{n} x_i = y_i$ iff $T \cup \Gamma \models x_i = y_i$, for some $i \in \{1, ..., n\}$. A Σ-theory T is *convex* iff all the conjunctions of Σ-literals are convex. A Σ-theory T is *stably-infinite* iff for any T-satisfiable Σ-formula φ, there exists a model of T whose domain is infinite and which satisfies φ. A *Nelson-Oppen* theory (**NO**-theory, for short) is a stably-infinite theory which admits a satisfiability algorithm. A **NOconvex**-theory is a convex **NO**-theory. The class of **NO**-theories (resp. **NOconvex**-theories) is denoted by **NO** (resp. **NOconvex**).

Theorem 1. **NO** *and* **NOconvex** *are closed under disjoint-union.*

Definition 2. A *solver* (denoted by *solve*) for a Σ-theory T is a function which takes as input a Σ-equality $s = t$ and such that (a) $solve(s = t)$ returns $false$, if $T \models s \neq t$, or (b) $solve(s = t)$ returns a substitution $\lambda = \{x_1 \leftarrow t_1, ..., x_n \leftarrow t_n\}$ such that (b.1) x_i is a variable occurring in s or t for $i = 1, ..., n$, (b.2) x_i does not occur in any t_j for $i, j = 1, ..., n$, and (b.3) $T \models s = t \Leftrightarrow \exists y_1, ..., y_m . \bigwedge_{i=1}^{n} x_i = t_i$, where $y_1, ..., y_m$ ($m \geq 0$) are "fresh" variables s.t. y_k does not occur in s or t, for all $k = 1, ..., m$. A conjunction of Σ-equalities is in *solved form* iff it has the form $\bigwedge_{i=1}^{n} x_i = t_i$, which will be denoted by $\hat{\lambda}$, where $\lambda = \{x_1 \leftarrow t_1, ..., x_n \leftarrow t_n\}$ is the substitution returned by *solve*. A *canonizer canon* for a Σ-theory T is an idempotent function from Σ-terms to Σ-terms such that $T \models a = b$ iff $\models canon(a) = canon(b)$. A *Shostak* theory is a convex theory which admits a solver and a canonizer. A **SH**-theory is a stably-infinite Shostak theory. The class of **SH**-theories is denoted by **SH**.

We assume **SH**-theories to be stably-infinite since this is necessary to combine them with other theories as suggested by many recent papers (see e.g. [18]). This

is not too restrictive since, as shown in [3], any convex theory with no trivial models is stably-infinite.

Proposition 1. [18] **SH** $\subseteq$ **NOconvex** $\subseteq$ **NO**.

3 Rational Reconstruction of Combination Schemas

Let T_i be a Σ_i-theory ($i = 1, 2$) such that $\Sigma_1 \cap \Sigma_2 = \emptyset$. We consider the problem of building a satisfiability procedure for $T_1 \cup T_2$. As a preliminary step, we consider a *purification process* converting any conjunction Φ of $\Sigma_1 \cup \Sigma_2$-literals into a conjunction of pure literals. Such a process is achieved by replacing each alien subterm t by a new variable x and adding the equality $x = t$ to Φ. This mechanism, called *variable abstraction*, is repeatedly applied to Φ until no more alien subterms t can be abstracted away. Obviously, the purification process always terminates yielding $\Phi_1 \wedge \Phi_2$, where Φ_i is a conjunction of Σ_i-literals ($i = 1, 2$) such that $\Phi_1 \wedge \Phi_2$ is equisatisfiable to Φ in $T_1 \cup T_2$. In the rest of this paper, without loss of generality, we consider the satisfiability of formulae of the form $\Phi_1 \wedge \Phi_2$ (or, equivalently, of configurations $\Phi_1; \Phi_2$), where Φ_i is a conjunction of i-pure literals.

Our combination schemas are specified by inference systems. To prove that an inference system R yields a satisfiability procedure, we follow a three steps methodology. First, we show that the derivation relation $\vdash_{\mathsf{R}}$ induced by R is terminating. Second, we prove that $\vdash_{\mathsf{R}}$ preserves (un-)satisfiability. Finally, we check that the normal forms defined by $\vdash_{\mathsf{R}}$ (i.e. configurations to which no rule in R can be applied) distinct from $false$ must be satisfiable. The proof of the last step proceeds by contradiction showing that a normal form distinct from $false$ cannot be unsatisfiable by using the following (technical) lemma from which the proof of correctness of Nelson-Oppen schema in [25] essentially depends.

Lemma 1. [25] If T_1 and T_2 are two signature-disjoint stably-infinite theories, then any conjunction $\Phi_1 \wedge \Phi_2$ of pure quantifier-free formulas is $T_1 \cup T_2$-satisfiable if and only if there exists some *identification* of shared variables in $Var(\Phi_1) \cap Var(\Phi_2)$—i.e. an idempotent substitution ξ from variables to variables—such that $\Phi_i \xi \wedge \xi_{\neq}$ is T_i-satisfiable for $i = 1, 2$, where $\xi_{\neq}$ is the formula $\bigwedge_{\{(x,y) \mid x\xi \neq y\xi\}} x \neq y$.

3.1 Combining Theories in NOconvex

We assume that T_1 and T_2 are in **NOconvex**. This implies the availability of two satisfiability procedures for T_1 and T_2. We consider the inference system NO obtained as the union of NO_1 presented in Figure 1 and NO_2 obtained from NO_1 by symmetry.[2] NO takes configurations of the form $\Phi_1; \Phi_2$ where Φ_i is a set of Σ_i-literals ($i = 1, 2$). Rule $\mathsf{Contradiction}_1$ reports the T_1-unsatisfiability of Φ_1 (and hence of $\Phi_1 \wedge \Phi_2$), detected by the available satisfiability procedure.

[2] A symmetric rule for T_2 is obtained from a rule for T_1 by swapping indexes 1 and 2. A symmetric inference system for T_2 is the set of symmetric rules for T_2 obtained from the rules for T_1.

$$\mathsf{Contradiction}_1 \quad \Phi_1;\Phi_2 \vdash false \qquad \text{if } \Phi_1 \text{ is } T_1\text{-unsatisfiable}$$

$$\mathsf{Deduction}_1 \quad \Phi_1;\Phi_2 \vdash \Phi_1;\Phi_2 \cup \{x = y\} \text{ if } \begin{cases} \Phi_1 \textit{ is } T_1\text{-satisfiable}, \\ \Phi_1 \wedge x \neq y \text{ is } T_1\text{-unsatisfiable}, \\ \Phi_2 \wedge x \neq y \text{ is } T_2\text{-satisfiable, and} \\ x, y \in Var(\Phi_1) \cap Var(\Phi_2) \end{cases}$$

Fig. 1. The Inference System NO_1

Rule $\mathsf{Deduction}_1$ propagates equalities between shared variables known to the procedure for T_1 to that for T_2 (if they are not already known to the latter). The problem of checking whether the equality $x = y$ is a logical consequence of $T_1 \cup \Phi_1$ is transformed into the problem of checking the T_1-unsatisfiability of $\Phi_1 \cup \{x \neq y\}$ so to be able to exploit the available satisfiability procedure.

Theorem 2. *Let T_1, T_2 be two signature-disjoint* **NOconvex***-theories. Let* NO *be the inference system defined as the union $\mathsf{NO}_1 \cup \mathsf{NO}_2$, where NO_1 is depicted in Figure 1 and NO_2 is obtained from NO_1 by symmetry. The relation $\vdash^*_{\mathsf{NO}}$ is terminating and $\Phi_1;\Phi_2 \vdash^*_{\mathsf{NO}} false$ iff $\Phi_1 \wedge \Phi_2$ is $T_1 \cup T_2$-unsatisfiable.*

Indeed, NO specifies only the essence of the Nelson-Oppen schema. Such a schema can be refined to increase efficiency. For example, the satisfiability procedures of some theories, such as Linear Arithmetic, can be extended so to derive entailed equalities while checking for satisfiability (see, e.g. [16, 26]) thereby avoiding the guessing done when applying $\mathsf{Deduction}_1$. In this paper, we will not consider this kind of amelioration (the interested reader is referred to [8] for a comprehensive guide-line to the efficient implementation of the Nelson-Oppen schema). In the following, we will consider refinements of NO which allow us to incorporate solvers and canonizers for theories in **SH**.

3.2 Combining Theories in SH

We assume that T_1 and T_2 are in **SH**. This implies the availability of a canonizer $canon_i$ and a solver $solve_i$ for each theory T_i $(i = 1, 2)$.

Preliminary to the combination schema, we extend solvers (cf. Definition 2) to handle sets of equalities as follows: $solve(\emptyset)$ returns the identity substitution ϵ; $solve(\Gamma \cup \{s = t\}) = false$, if $solve(s = t) = false$; and $solve(\Gamma \cup \{s = t\}) = \sigma \circ solve(\Gamma\sigma)$, if $solve(s = t) = \sigma$, where $\circ$ denotes composition of substitutions.

We consider the inference system SH obtained as the union of SH_1 presented in Figure 2 and SH_2 obtained from SH_1 by symmetry. SH takes configurations of the form $\Gamma_1, \Delta_1; \Gamma_2, \Delta_2$, where Γ_i is a set of Σ_i-equalities and Δ_i is a set of Σ_i-disequalities for $i = 1, 2$. Rule $\mathsf{Solve} - \mathsf{fail}_1$ reports the T_1-unsatisfiability of Γ_1 (and hence of $\Gamma_1 \wedge \Delta_1 \wedge \Gamma_2 \wedge \Delta_2$) detected by $solve_1$. Rule $\mathsf{Solve} - \mathsf{success}_1$ replaces the Σ_1-equalities Γ_1 with their solved form which is obtained again by using $solve_1$. This is important for the next two rules. Dealing with solved

$$\mathsf{Solve}-\mathsf{fail}_1 \quad \Gamma_1, \Delta_1; \Gamma_2, \Delta_2 \vdash \mathit{false} \quad \text{if } \mathit{solve}_1(\Gamma_1) = \mathit{false}$$

$$\mathsf{Solve}-\mathsf{success}_1 \quad \Gamma_1, \Delta_1; \Gamma_2, \Delta_2 \vdash \widehat{\gamma_1}, \Delta_1; \Gamma_2, \Delta_2 \text{ if } \begin{cases} \Gamma_1 \text{ is not in solved form,} \\ \gamma_1 = \mathit{solve}_1(\Gamma_1) \neq \mathit{false} \end{cases}$$

$$\mathsf{Contradiction}_1 \quad \widehat{\gamma_1}, \Delta_1 \cup \{s \neq t\}; \Gamma_2, \Delta_2 \vdash \mathit{false} \quad \text{if } \mathit{canon}_1(s\gamma_1) = \mathit{canon}_1(t\gamma_1)$$

$$\mathsf{Deduction}_1 \quad \widehat{\gamma_1}, \Delta_1; \widehat{\gamma_2}, \Delta_2 \vdash \widehat{\gamma_1}, \Delta_1; \widehat{\gamma_2} \cup \{x = y\}, \Delta_2 \text{ if } \begin{cases} \mathit{canon}_1(x\gamma_1) = \mathit{canon}_1(y\gamma_1), \\ \mathit{canon}_2(x\gamma_2) \neq \mathit{canon}_2(y\gamma_2), \\ x, y \in \mathit{Var}(\gamma_1) \cap \mathit{Var}(\gamma_2) \end{cases}$$

Fig. 2. The Inference System SH_1

forms allows us to simply determine entailed equalities (possibly between shared variables, see $\mathsf{Deduction}_1$) using canonizers. Hence, it is possible to lazily report unsatisfiability as soon as we find a disequality whose corresponding equality is entailed (see $\mathsf{Contradiction}_1$). Indeed, convexity allows us to handle disequalities one by one.

Theorem 3. *Let T_1, T_2 be two signature-disjoint* **SH***-theories. Let* SH *be the inference system defined as the union* $\mathsf{SH}_1 \cup \mathsf{SH}_2$*, where* SH_1 *is depicted in Figure 2 and* SH_2 *is obtained by symmetry. The relation* $\vdash^*_{\mathsf{SH}}$ *is terminating and* $\Gamma_1, \Delta_1; \Gamma_2, \Delta_2 \vdash^*_{\mathsf{SH}} \mathit{false}$ *iff* $\Gamma_1 \wedge \Delta_1 \wedge \Gamma_2 \wedge \Delta_2$ *is* $T_1 \cup T_2$*-unsatisfiable.*

It is easy to see that a strategy applying rules $\mathsf{Solve}-\mathsf{fail}_1$, $\mathsf{Solve}-\mathsf{success}_1$, and $\mathsf{Contradiction}_1$ in SH to a configuration $\Gamma_1, \Delta_1; \Gamma_2, \Delta_2$ yields the same result as that of applying rule $\mathsf{Contradiction}_1$ in NO to $\Gamma_1 \cup \Delta_1; \Gamma_2 \cup \Delta_2$. Similarly, the application of rules $\mathsf{Solve}-\mathsf{success}_1$ and $\mathsf{Deduction}_1$ in SH simulates the application of $\mathsf{Deduction}_1$ in NO; showing that equalities between shared variables can be derived by invoking a solver (and a canonizer) rather than resorting to guessing as for NO when applying the rule $\mathsf{Deduction}_i$ $(i = 1, 2)$. This is one of the key insights underlying Shostak schema. Furthermore, similarly to [13], the abstract schema presented here seems to emphasize the importance of the solver w.r.t. the canonizer. In fact, if the solved form returned by the solver is also canonical, the canonizer can be trivially implemented as the identity function. Nonetheless, we believe that the concept of canonizer is quite important for two crucial reasons. First, it offers the entry point to refinements of the proposed schema to increase efficiency. In fact, solving a set of equalities in "one-shot", as done when applying rule $\mathsf{Solve}-\mathsf{success}_1$, may not be as efficient as solving equalities incrementally, as done e.g. in [22, 14]. This can be incorporated in our schema by refining the inference system SH along the lines described in [4] so that the solver is applied to only one equality at a time and the canonizer needs to return a canonical form for arbitrary terms. The second reason is that a generalization of the concept of canonizer will be the basis for a new combination schema as we will see in Section 4.

3.3 Combining a Theory in NOconvex with One in SH

Without loss of generality, let us assume that T_1 is in **NOconvex** and that T_2 is in **SH**. This situation frequently arises in practical verification problem, e.g. the union of a theory in **SH** and $\mathcal{E}$ (which is *not* in **SH**). We consider the inference system NS obtained as the union of NO_1 in Figure 1 and SH_2, the symmetric of SH_1 in Figure 2. NS takes configurations of the form $\Phi_1; \Gamma_2, \Delta_2$ where Φ_1 is a set of Σ_1-literals, Γ_2 is a set of Σ_2-equalities, and Δ_2 is a set of Σ_2-disequalities. We furtherly assume that when a rule of NO is applied, $\Phi_1; \Gamma_2, \Delta_2$ stands for $\Phi_1; \Gamma_2 \cup \Delta_2$ and when a rule of SH is applied, $\Phi_1; \Gamma_2, \Delta_2$ is considered as $\Gamma_1, \Delta_1; \Gamma_2 \cup \Delta_2$, where $\Phi_1 = \Gamma_1 \cup \Delta_1$ and Γ_1 (Δ_1) is a set of Σ_1-equalities (-disequalities, respectively). NS can be seen as an abstract version of the one proposed in [3].

Theorem 4. *Let T_1, T_2 be two signature-disjoint theories such that T_1 is in* **NOconvex** *and T_2 is in* **SH**. *Let NS be the inference system defined as the union $\mathsf{NO}_1 \cup \mathsf{SH}_2$, where NO_1 is in Figure 1 and SH_2 is obtained from SH_1 in Figure 2 by symmetry. The relation $\vdash^*_{\mathsf{NS}}$ is terminating and $\Phi_1; \Gamma_2, \Delta_2 \vdash^*_{\mathsf{NS}} false$ iff $\Phi_1 \wedge \Gamma_2 \wedge \Delta_2$ is $T_1 \cup T_2$-unsatisfiable.*

Let $T_1, ..., T_k$ and $T_{k+1}, ..., T_{k+n}$ be k theories in **NOconvex** and n theories in **SH**, respectively, and such that $\Sigma_i \cap \Sigma_j \neq \emptyset$ for $i, j = 1, \ldots, k+n$, $i \neq j$, and $n, k \geq 1$. It is possible to modularly build a satisfiability procedure for $T = \bigcup_{j=1}^{k+n} T_j$ as follows. Repeatedly use NO to obtain a satisfiability procedure for $U_0 = \bigcup_{j=1}^{k} T_j$, then repeatedly use NS to build satisfiability procedures for $U_1 = U_0 \cup T_{k+1}, \ldots, U_n = U_{n-1} \cup T_{k+n}$, where U_n is T. An alternative would be to repeatedly use SH to construct satisfiability procedures for unions of two theories in **SH**, followed by a repeated use of NO on the resulting theories. The particular case of combining $\mathcal{E}$ with one or more theories in **SH** (i.e. $k = 1$) has been extensively studied by many researchers [5, 22, 14, 3, 13, 17, 23, 4], Shostak [24] being the first. The correctness of the combination schemas outlined above immediately follows from the correctness of NO, SH, NS, the fact that the union of two theories in **NOconvex** is also in **NOconvex** (Theorem 1), and that **SH** is contained in **NOconvex** (Proposition 1). Similar results are given in [18]. Finally, let us mention still another possibility to combine k theories in **NOconvex** and n theories in **SH**. It would be possible to slightly modify our inference rules to take into account $k + n$ theories; configurations would be of the form $\Phi_1; \ldots; \Phi_k; \Gamma_{k+1}, \Delta_{k+1}; \ldots; \Gamma_{k+n}, \Delta_{k+n}$ and the rule $\mathsf{Deduction}$ would propagate an equality between shared variables, deduced in one theory, to the other $(k + n) - 1$ theories. At this point, it would not be difficult to modify the proof of correctness for NS to show that the resulting rules (taken from $\mathsf{NO}_1, \ldots, \mathsf{NO}_k, \mathsf{SH}_{k+1}, \ldots, \mathsf{SH}_{k+n}$) yield a satisfiability procedure for T. The resulting proof would be a bit more involved because of the more complex notation.

4 Combining ECANconvex-Theories

Although the combination schemas of Section 3 are already sufficient to combine several theories either in **NOconvex**, **SH**, or both, we investigate how to find a generic combination schema which features the modularity of NO and retains some of the efficiency of SH. To this end, we introduce a new basic building block which generalizes the concept of canonizer for **SH**-theories and can be modularly combined either to (1) build a satisfiability procedure for the union of theories (admitting extended canonizers) by a schema which allows to efficiently propagate entailed equalities as in SH but does not require to solve equalities, or to (2) obtain an extended canonizer out of two extended canonizers in a modular way, thereby showing that the class of theories for which an extended canonizer exists is closed under disjoint union.

4.1 Extended Canonizers and ECANconvex-Theories

Definition 3. Let T be a Σ-theory with decidable uniform word problem, and let Γ be a conjunction of Σ-equalities. Given any T-satisfiable Γ, an *extended canonizer* for T is a function $ecan(\Gamma) : T(\Sigma, X) \to T(\Sigma \cup K(\Gamma), X)$, such that, for any terms s, t, we have $T \models \Gamma \Rightarrow s = t$ iff $ecan(\Gamma)(s) = ecan(\Gamma)(t)$, where $K(\Gamma)$ is a finite set of fresh constant symbols such that $\Sigma \cap K(\Gamma) = \emptyset$.

ECANconvex denotes the class of convex theories admitting an extended canonizer.

The concept of extended canonizer presents many similarities with the function $can(\Gamma)$ in [22].[3] An important difference is that our extended canonizers can be modularly combined to yield satisfiability procedures for union of disjoint theories (see Section 4.3 below). However, [23] describes a solution to the problem of combining $\mathcal{E}$ with several theories in **SH** by means of an interesting generalization of Shostak algorithm which only requires to build a canonizer for the union of the theories (which is always possible for convex theories [17]) and invokes the solvers for the theories being combined.

If a theory T admitting an extended canonizer *ecan* is also convex, then it is always possible to build a satisfiability procedure for T by recalling that $\Gamma \wedge \neg e_1 \wedge \cdots \wedge \neg e_n$ is T-unsatisfiable if and only if there exists some $i \in \{1, \ldots, n\}$ such that $\Gamma \wedge \neg e_i$ is T-unsatisfiable, or equivalently $T \models \Gamma \Rightarrow e_i$. This immediately implies the following proposition.

Proposition 2. ECANconvex $\subseteq$ NOconvex.

Although we can always decide the uniform word problem for a convex theory T by invoking a satisfiability procedure, it is not clear whether an extended canonizer always exists for T in **NOconvex**. Recall that in the Definition 3 of

[3] $can(\Gamma)(t)$ returns a canonical form of the term t in which any (uninterpreted) subterm that is congruent to a known left hand side in an equation of Γ is replaced by the associated right hand side.

extended canonizer, we require it to return terms over $T(\Sigma \cup K(\Gamma), X)$ where $K(\Gamma)$ must be a *finite* set of fresh constant symbols. The intuition is that a constant in $K(\Gamma)$ is the representative of an equivalence class induced by $T \cup \Gamma$. Since $K(\Gamma)$ is finite, extended canonizers can only be built for a theory T such that, for each finite set Γ of ground equalities, the equivalence relation induced by $T \cup \Gamma$ has a finite number of equivalence classes. So, the problem of determining that the inclusion in Proposition 2 is strict amounts to proving the existence of a theory T in **NOconvex** such that for some set Γ of ground equalities, $T \cup \Gamma$ induces an equivalence relation with an infinite number of equivalence classes. We conjecture that such a theory exists and hence conclude the inclusion in Proposition 2 is strict.

4.2 Extended Canonizers, Solvers, Canonizers, and Satisfiability Procedures

We describe the relationships between theories in **ECANconvex**, those in **SH**, and some in **NOconvex** which are not in **SH**.

Proposition 3. **SH** $\subseteq$ **ECANconvex**, i.e. theories in **SH** admits an extended canonizer.

Proof. Let T be an **SH**-theory and *solve* and *canon* its solver and canonizer, respectively. We define an extended canonizer *ecan* for T, as follows. If $solve(\Gamma) = false$, then *ecan* is undefined. If $solve(\Gamma)$ returns a substitution λ, then $ecan(\Gamma)(s)$ returns $canon(s\lambda)$. □

The proof of the Proposition above suggests how to reduce the uniform word problem $T \models \Gamma \Rightarrow s = t$ for a theory T in **SH** to the word problem $T \models s\lambda = t\lambda$, where λ is the substitution obtained by iteratively applying *solve* to Γ. In turn, $T \models s\lambda = t\lambda$ reduces to checking whether $canon(s\lambda)$ is syntactically equal to $canon(t\lambda)$ (a similar observation is done in [13]). The key observation here is that substituting equalities in Γ with their solved form $\hat{\lambda}$ can be done without backtracking thanks to the properties of *solve*. This is not possible for some theories whose uniform word problem can be decided by using an extended canonizer. For example, $\mathcal{E}$ admits efficient algorithms to solve its uniform word problem (see, e.g. [9]) but it is easy to show that it does not admit a solver (see e.g. [18]); so $\mathcal{E}$ is not in **SH**.

Proposition 4. $\mathcal{E} \in$ **ECANconvex**, i.e. $\mathcal{E}$ admits an extended canonizer.

The extended canonizer for $\mathcal{E}$ is a total function since any set Γ of ground equalities is $\mathcal{E}$-satisfiable. Because of Proposition 4 and the fact that $\mathcal{E}$ is not in **SH**, the inclusion between **SH** and **NOconvex** in Proposition 3 is strict. There are other interesting theories not in **SH** for which an extended canonizer exists as the following Proposition shows.

Proposition 5. Let C_f be the theory axiomatized by $\forall X, Y. f(X, Y) = f(Y, X)$. Then, the theory $C'_f := C_f \cup \{\exists X, Y. X \neq Y\}$ is not in **SH** and admits an extended canonizer.

Also, associative-commutative theories can be shown to admit extended canonizers by using the result in [2].

An efficient implementation of the uniform word problem for the theory of equality and commutative symbols based on a fast congruence closure algorithm is given in [9]. This can be used as the basis for efficient extended canonizers.

4.3 Extended Canonizers and Combination of ECANconvex-Theories

For technical reasons (that will become clear in a moment), we introduce the concept of *equational simplifier*, which is a partial function *eqs* taking conjunctions of equalities and returning a function whose input is an equality and which returns either *true* or $false$ such that for any conjunction of equalities Γ and equality e, (a) *eqs* is defined for Γ and e iff Γ is T-satisfiable, and (b) $eqs(\Gamma)(e)$ is *true* if $T \models \Gamma \Rightarrow e$, and $false$ otherwise. Indeed, for T in **ECANconvex**, clause (b) can be restated as follows: for any T-satisfiable Γ and any equality $s = t$, $eqs(\Gamma)(s = t) = true$ iff $ecan(\Gamma)(s) = ecan(\Gamma)(t)$. In the rest of this section, we assume that equational simplifiers are defined in terms of extended canonizers and we study the problem of building satisfiability procedures for unions of **ECANconvex**-theories. There are two possibilities. First, adapt NO to combine satisfiability procedures built out of equational simplifiers. (To see this, observe that equational simplifiers decides uniform word problems since these can be transformed to satisfiability problems as described in Section 4.1.) This gives a straightforward reformulation of the inference rules in NO where side conditions are expressed in terms of *eqs*. Since this is easy, the details are left to the reader. Second, build an equational simplifier for the union of theories and then derive the corresponding satisfiability procedure. In the following, we develop the second possibility. Let T_i be a Σ_i-theory in **ECANconvex** and $ecan_i$ its extended canonizer for $i = 1, 2$ and $\Sigma_1 \cap \Sigma_2 = \emptyset$. First, we show how to obtain an equational simplifier $eqs_{1,2}$ for $T_1 \cup T_2$ by using a variant of NO restricted to equalities. Then, we show how to build an extended canonizer $ecan_{1,2}$ for $T_1 \cup T_2$ out of $ecan_1, ecan_2$ and $eqs_{1,2}$. The reader may ask why we need to build the equational simplifier for $T_1 \cup T_2$ to be able to build an extended canonizer. The answer is in the definition of extended canonizer which requires Γ to be satisfiable for $ecan(\Gamma)$ to be defined. So, we need to check the $T_1 \cup T_2$-satisfiability of conjunctions of $\Sigma_1 \cup \Sigma_2$-equalities to decide whether $ecan_{1,2}$ is defined.

Lemma 2. Let T_1 and T_2 be two signature-disjoint convex and stably infinite theories. If an equational simplifier eqs_i is known for T_i (for $i = 1, 2$), then it is possible to construct an equational simplifier *eqs* for $T_1 \cup T_2$ using the combination method described in Figure 3.

Notice that the result above can be repeatedly applied to build an equational simplifier for the union of n signature-disjoint, convex, and stably-infinite theories $T_1, \ldots, T_n$. So, a satisfiability procedure for $T_1 \cup \cdots \cup T_n$ can be immediately obtained. However, this still does not answer the question: does there exist an extended canonizer $ecan_{1,2}$ for $T_1 \cup T_2$ given two extended canonizers

Given a set Γ of equalities and an equality $s = t$, the following procedure shows how to construct eqs for $(\Gamma, s = t)$, when defined. Let EEC be the inference system defined as the union $\mathsf{EEC}_1 \cup \mathsf{EEC}_2$, where EEC_1 is depicted in Figure 4 and EEC_2 is obtained by symmetry.

1. Purify Γ into $\Gamma_1; \Gamma_2$.
2. If $\Gamma_1; \Gamma_2 \vdash^*_{\mathsf{EEC}} false$, then eqs is undefined for $(\Gamma, \_)$.
3. Otherwise, let $\Gamma'_1; \Gamma'_2$ be the normal form w.r.t. $\vdash_{\mathsf{EEC}}$ such that $\Gamma_1; \Gamma_2 \vdash^*_{\mathsf{EEC}} \Gamma'_1; \Gamma'_2$. Furthermore, purify $x = s \wedge y = t$, where x, y are new variables not occurring in $Var(\Gamma'_1 \wedge \Gamma'_2)$. Let $\Gamma''_1; \Gamma''_2$ be the result of purifying $\Gamma'_1 \wedge \Gamma'_2 \wedge x = s \wedge y = t$.
4. Let $\Gamma'''_1; \Gamma'''_2$ be the normal form w.r.t. $\vdash_{\mathsf{EEC}}$ such that $\Gamma''_1; \Gamma''_2 \vdash^*_{\mathsf{EEC}} \Gamma'''_1; \Gamma'''_2$. This normal form is necessarily different from $false$ since $\Gamma_1 \wedge \Gamma_2$ is $T_1 \cup T_2$-satisfiable and x, y are different new variables.
5. If there exists $i \in \{1, 2\}$ such that $x, y \in Var(\Gamma''_i)$, then $eqs(\Gamma)(s = t)$ is defined and it is equal to $eqs_i(\Gamma'''_i)(x = y)$.
6. Otherwise ($x \in Var(\Gamma''_i), y \in Var(\Gamma''_j)$, for $i \neq j$), $eqs(\Gamma)(s = t)$ is defined, and it is equal to $true$ if there exists $z \in Var(\Gamma''_1) \cap Var(\Gamma''_2)$ such that $eqs_i(\Gamma'''_i)(x = z) = eqs_j(\Gamma'''_j)(y = z) = true$, otherwise it is defined as $false$.

Fig. 3. Equational Simplifier for the Union of Theories

$ecan_1$ and $ecan_2$ for T_1 and T_2, respectively, and an equational simplifier $eqs_{1,2}$ for their union? To answer this question (constructively), we analyze the equational simplifier for $eqs_{1,2}$ for $T_1 \cup T_2$ given in Figure 3 and we show how an extended canonizer can be obtained. The key technique underlying the analysis consists of unfolding the fresh variables (abstracting alien subterms) introduced by purification so to get terms back in the right signature. This unfolding must be done with care since we must take into account the equivalence relation on fresh variables induced by the propagation of equalities between shared variables.

Theorem 5. ECANconvex *is closed under disjoint union.*

$$\mathsf{Contradiction}_1 \quad \Gamma_1; \Gamma_2 \vdash false \qquad \text{if } eqs_1(\Gamma_1) \text{ is undefined}$$

$$\mathsf{Deduction}_1 \quad \Gamma_1; \Gamma_2 \vdash \Gamma_1; \Gamma_2 \cup \{x = y\} \text{ if } \begin{cases} eqs_1(\Gamma_1) \text{ is defined,} \\ eqs_2(\Gamma_2) \text{ is defined,} \\ eqs_1(\Gamma_1)(x = y) = true, \\ eqs_2(\Gamma_2)(x = y) = false, \\ x, y \in Var(\Gamma_1) \cap Var(\Gamma_2) \end{cases}$$

Fig. 4. The Inference System EEC_1

5 Conclusions and Future Work

We have presented combination schemas for disjoint unions of (a) two theories in **NOconvex**, (b) two theories in **SH**, and (c) one theory in **NOconvex** with one in **SH**. We have shown how such schemas are related to Nelson-Oppen and Shostak approaches to combination as well as with many of the refinements available in the literature. Our formalization highlights the key ideas underlying each combination and allows us to derive proofs of correctness which are easy to grasp. We believe this is a valuable synthesis for further investigations. To justify this claim, we have introduced the concept of extended canonizer which abstracts algorithms for deciding the uniform word problem of a theory and it is modular, i.e. an extended canonizer can be built out of the extended canonizers for the component theories. This is in contrast to the lack of modularity of solvers for Shostak combination schema. Another advantage is the fact that it can be easily implemented in terms of solvers and canonizers for Shostak theories or by rewriting techniques as suggested e.g. in [1].

There are several main lines for future work. First, we want to derive a more precise characterization of the theories admitting an extended extended canonizer. In this respect, a promising line of research would be to study for which theories the uniform word problem can be reduced to a word problem. Second, we want to study the complexity of extended canonizers in the union of theories. We believe it would be interesting to apply our combination results to polynomial time decidable uniform word problems as described in [12]. Third, we intend to empirically evaluate the efficiency of extended canonizers by conducting some experiments in haRVey [6]. The interest here is to obtain an efficient combination between extended canonizers and propositional solvers. This requires to equip extended canonizers with the capability of generating useful theory-specific facts which, once projected into the propositional domain, allow to reduce the search space. Finally, we plan to study how extended canonizers can be used when non-convex theories are combined.

References

1. A. Armando, S. Ranise, and M. Rusinowitch. A Rewriting Approach to Satisfiability Procedures. *Info. and Comp.*, 183(2):140–164, June 2003.
2. L. Bachmair, A. Tiwari, and L. Vigneron. Abstract Congruence Closure. *Journal of Automated Reasoning*, 31(2):129–168, 2003.
3. C. W. Barrett, D. L. Dill, and A. Stump. A generalization of Shostak's method for combining decision procedures. In *Proc. of the 4th Int. Workshop on Frontiers of Combining Systems*, volume 2309 of *LNCS*, pages 132–147, 2002.
4. S. Conchon and S. Krstić. Strategies for combining decision procedures. In *Proc. of the 9th Int. Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 2619 of *LNCS*, pages 537–553. Springer-Verlag, April 2003.
5. D. Cyrluk, P. Lincoln, and N. Shankar. On Shostak's decision procedure for combinations of theories. In *Proc. of the 13th Int. Conference on Automated Deduction*, volume 1104 of *LNCS*, pages 463–477. Springer-Verlag, 1996.

6. D. Déharbe and S. Ranise. Light-Weight Theorem Proving for Debugging and Verifying Units of Code. In I. C. S. Press, editor, *Proc. of the Int. Conf. on Software Engineering and Formal Methods (SEFM03)*, 2003.
7. N. Dershowitz and J.-P. Jouannaud. *Handbook of Theoretical Computer Science*, volume B, chapter 6: Rewrite Systems, pages 244–320. 1990.
8. D. Detlefs, G. Nelson, and J. B. Saxe. Simplify: A Theorem Prover for Program Checking. Technical Report HPL-2003-148, HP Laboratories, 2003.
9. P. J. Downey, R. Sethi, and R. E. Tarjan. Variations on the Common Subexpression Problem. *J. of the ACM*, 27(4):758–771, October 1980.
10. H. B. Enderton. *A Mathematical Introduction to Logic*. Ac. Press, Inc., 1972.
11. J.-C. Filliâtre, S. Owre, H. Rueß, and N. Shankar. ICS: Integrated Canonization and Solving (Tool presentation). In *Proc. of CAV'2001*, volume 2102 of *LNCS*, pages 246–249. Springer-Verlag, 2001.
12. H. Ganzinger. Relating semantic and proof-theoretic concepts for polynomial time decidability of uniform word problems. In *Proc. 16th IEEE Symp. on Logic in Computer Science*, pages 81–92. IEEE Comp. Soc. Press, 2001.
13. H. Ganzinger. Shostak light. In *Proc. of the 18th Int. Conference on Automated Deduction*, volume 2392 of *LNCS*, pages 332–346. Springer-Verlag, jul 2002.
14. D. Kapur. A rewrite rule based framework for combining decision procedures. In *Proc. of the 4th Int. Workshop on Frontiers of Combining Systems*, volume 2309 of *LNCS*, pages 87–102. Springer-Verlag.
15. D. Kapur. Shostak's congruence closure as completion. In *Proc. of the 8th Int. Conference on Rewriting Techniques and Applications*, volume 1232 of *LNCS*. Springer-Verlag, 1997.
16. D. Kapur and X. Nie. Reasoning about Numbers in Tecton. In *Proc.* 8^{th} *Inl. Symp. Methodologies for Intelligent Systems*, pages 57–70, 1994.
17. S. Krstić and S. Conchon. Canonization for disjoint unions of theories. In *Proc. of the 19th Int. Conference on Automated Deduction*, volume 2741 of *LNCS*, Miami Beach, FL, USA, 2003. Springer Verlag.
18. Z. Manna and C. G. Zarba. Combining decision procedures. In *Formal Methods at the Cross Roads: From Panacea to Foundational Support*, volume 2757 of *LNCS*, pages 381–422. Springer, 2003.
19. G. Nelson and D. C. Oppen. Simplification by cooperating decision procedures. *ACM Trans. on Programming Languages and Systems*, 1(2):245–257, Oct. 1979.
20. D. C. Oppen. Complexity, convexity and combinations of theories. *Theoretical Computer Science*, 12:291–302, 1980.
21. S. Ranise, C. Ringeissen, and D.-K. Tran. Nelson-Oppen, Shostak and the Extended Canonizer: A Family Picture with a Newborn (Full Version). Available at `http://www.loria.fr/~ranise/pubs/long-ictac04.ps.gz`.
22. H. Rueß and N. Shankar. Deconstructing Shostak. In *Proceedings of the 16th Annual IEEE Symposium on Logic in Computer Science*, pages 19–28. IEEE Computer Society, June 2001.
23. N. Shankar and H. Rueß. Combining shostak theories. In *Proc. of the 13th Int. Conf. on Rewriting Techniques and Applications*, volume 2378 of *LNCS*, pages 1–18. Springer, 2002.
24. R. E. Shostak. Deciding combinations of theories. *J. of the ACM*, 31:1–12, 1984.
25. C. Tinelli and C. Ringeissen. Unions of non-disjoint theories and combinations of satisfiability procedures. *Theoretical Computer Science*, 290(1):291–353, 2003.
26. P. van Hentenryck and T. Graf. Standard Forms for Rational Linear Arithmetics in Constraint Logic Programming. *Annals of Mathematics and Artificial Intelligence*, 5:303–319, 1992.