On unification for bounded distributive lattices

Viorica Sofronie-Stokkermans

Max-Planck-Institut für Informatik, Stuhlsatzenhausweg 85, Saarbrücken, Germany

email: sofronie@mpi-sb.mpg.de

We give a method for deciding unifiability in the variety of bounded distributive lattices. For this, we reduce the problem of deciding whether a unification problem $\mathcal{S}$ has a solution to the problem of checking the satisfiability of a set $\Phi_{\mathcal{S}}$ of ground clauses. This is achieved by using a structure-preserving translation to clause form. The satisfiability check can then be performed either by a resolution-based theorem prover or by a SAT checker. We apply the method to unification with free constants and to unification with linear constant restrictions, and show that, in fact, it yields a decision procedure for the positive theory of the variety of bounded distributive lattices. We also consider the problem of unification over (i.e. in an algebraic extension of) the free lattice. Complexity issues are also addressed.

Categories and Subject Descriptors: F.4.1 [**Mathematical Logic**]: ; I.2.3 [**Deduction and Theorem Proving**]: ; I.1.1 [**Symbolic and algebraic manipulation**]:

1. INTRODUCTION

From an algebraic point of view, unification can be seen as solving systems of equations in the initial or free algebra of an equational theory. Apart from its theoretical interest, unification is used e.g. in resolution-based theorem proving and in term rewriting to deal with certain equational axioms, such as associativity and commutativity. The unification problem has been thoroughly studied for equationally defined theories characterized by axioms such as associativity, commutativity, distributivity, associativity-commutativity, associativity-commutativity-idempotency; and for several theories related to algebra (abelian groups, commutative and Boolean rings, semilattices, Boolean algebras, primal algebras, discriminator varieties). For details cf. [Baader and Snyder 2001] and the bibliography cited there. The combination of unification algorithms has been studied in [Baader and Schulz 1998].

In this paper we present some results on unification in the equational theory of bounded distributive lattices. We study the decidability, and give decision procedures for the following problems:

—unification with and without constants in the class D_{01} of bounded distributive lattices,

—unification with linear constant restrictions in D_{01},
—the positive theory of D_{01},
—unification *over* (i.e. in an algebraic extension of) the free distributive lattice with n generators.

The study was motivated, on the one hand, by our interest in distributive lattices with operators, and, on the other hand, by the fact that unification problems in semilattice- and lattice-based structures are becoming of increasing interest in computer science. We mention, for instance, the results of Baader and Narendran on unification of concept terms in description logics [Baader and Narendran 2001]; similar possible applications in set constraints may also be of interest.

Deciding whether a unification problem with respect to a certain class of algebraic structures has a solution may not always be satisfactory, especially in problems related to applications (such as, for instance unification of concept terms in description logics). Even if the unification problem has no solution, one might be interested to know whether it has a solution in some extension of the free algebra[1]. This is why we also study unification over the free distributive lattice.

It is known that the class D_{01} of bounded distributive lattices has an undecidable first-order theory (cf. e.g. [Burris and McKenzie 1981], page 16). The uniform word problem for distributive lattices is decidable (since $\mathsf{D}_{01} = ISP(2)$, where 2 is the 2-element lattice), and has been proved to be co-NP-hard by Bloniarz et al. [Hunt et al. 1987]. By a result of McKinsey [McKinsey 1943] it follows that the universal theory of the variety of distributive lattices is decidable. Also the positive $\forall\exists$ theory of D_{01} (hence the unification problem with free constants) is decidable. This is a consequence of the fact that the class D_{01} is locally finite (see also Section 3). Unification for distributive lattices has only been addressed in a few papers. In [Gerhard and Petrich 1994], Gerhard and Petrich give a criterion for unifiability (with free constants) of *two* terms in the theory of distributive lattices.[2] Then, in the attempt to give a basis set for all unifiers of two terms, they considered terms containing only one of the lattice operations $\vee$ or $\wedge$ and, for more general terms, only particular cases, containing few variables. The results of Ghilardi [Ghilardi 1997] show that the equational class D of distributive lattices has unification type 0, i.e. there exist D-unification problems with no minimal complete set of unifiers. We are not aware of any other results on unification for distributive lattices, e.g. concerning its complexity. Due to the interaction between operators, neither the ideas used in [Schmidt-Schauß 1998] for distributive unification, nor the results in [Baader and Schulz 1998] on the combination of unification algorithms can be applied in this case.

The method we propose in this paper uses the fact that the free distributive lattice with n generators can be represented as a lattice of upwards closed subsets (order filters) of a finite partially ordered set. Various types of representation theorems have already been used, at a theoretical level, for investigating free algebras, as

[1]This situation is similar to the problems which occur when trying to solve the equation $x^2+1 = 0$. This equation has no solution in $\mathbb{R}$, but it has the solutions i and $-i$ in the extension $\mathbb{C}$ of $\mathbb{R}$.
[2]We were not able to generalize the argument used in the proof of this result to handle conjunctions of equations.

well as existentially and algebraically closed members of several varieties of lattice-ordered algebras, but the idea of using them for obtaining decision procedures for unification problems is, to the best of our knowledge, new. We used such representation theorems for bounded distributive lattices in [Sofronie-Stokkermans 1999; 2003], and showed that they allow in a natural way to give resolution-based decision procedure for the *universal theory* of certain varieties of distributive lattices with operators. However, the arguments in [Sofronie-Stokkermans 1999] cannot be used for the positive $\forall\exists$ theory of such varieties without modification.

In this paper we show that the use of the Birkhoff representation for finite bounded distributive lattices allows to reduce the unification problem in D_{01} to the problem of checking the satisfiability of a set of ground clauses. Based on this idea, we propose an algorithm which consists of the following steps:

(1) *Structure-preserving translation to clause form:* testing the satisfiability of a unification problem $\mathcal{S}$ is reduced to the problem of checking the satisfiability of a set $\Phi_{\mathcal{S}}$ of ground clauses.
(2) Testing the satisfiability of $\Phi_{\mathcal{S}}$ can be done using any method for checking satisfiability of sets of clauses in propositional logic.

The main advantage of our approach is that the structure-preserving translation to clause form makes it much easier to treat the unification problem for bounded distributive lattices, by using classical methods for checking the satisfiability of sets of clauses.

We show that similar ideas can be used for unification with linear constant restrictions (for definitions cf. Section 2 or [Baader and Schulz 1998]). As a byproduct, using Prop. 5.6 in [Baader and Schulz 1996], our results show that standard methods for satisfiability checking can be used for deciding the positive theory of D_{01}. The paper extends and improves results presented in [Sofronie-Stokkermans 2000].

1.1 Idea

We illustrate the idea of the algorithm we propose on a simple example. Consider the following formula (which is not true in all distributive lattices):

$$\phi = \forall x_1 x_2 x_3 \exists y (x_1 \wedge y = x_2 \wedge x_3 \text{ and } x_1 \vee y = x_2 \vee x_3).$$

One possibility for showing that ϕ is not valid in D_{01} is to show that its negation is consistent with the bounded distributive lattice axioms. In order to check the satisfiability of the formulae obtained this way we can use any theorem prover for first-order logic with equality such as, for instance, SPASS [Weidenbach et al. 1996]. However, equational reasoning in the presence of lattice axioms is quite difficult, and it may be difficult or even impossible to set the parameters of SPASS in order to enforce termination.

Another possibility is to use the fact that $\mathsf{D}_{01} \models \phi$ iff $\exists y(c_1 \wedge y = c_2 \wedge c_3$ and $c_1 \vee y = c_2 \vee c_3)$ holds in the free algebra in D_{01} freely generated by $\{c_1, c_2, c_3\}$, denoted in what follows by $F_{\mathsf{D}_{01}}(\{c_1, c_2, c_3\})$, i.e. iff

$$F_{\mathsf{D}_{01}}(\{c_1, c_2, c_3\}) \models \exists y(c_1 \wedge y = c_2 \wedge c_3 \text{ and } c_1 \vee y = c_2 \vee c_3).$$

The straightforward way of checking this would be to analyze every instantiation of y in the algebra $F_{\mathsf{D}_{01}}(\{c_1, c_2, c_3\})$, which has 20 elements, and for each such instantiation we would need to decide two word problems, i.e. to check whether $c_1 \wedge y = c_2 \wedge c_3$ and $c_1 \vee y = c_2 \vee c_3$ are true under the instantiation.

Instead, we use the fact that $F_{\mathsf{D}_{01}}(\{c_1, c_2, c_3\})$ is isomorphic to a lattice of sets, more precisely to the lattice of all order-filters of $(\mathcal{P}(\{c_1, c_2, c_3\}), \subseteq)$; the isomorphism maps each generator $c \in C$ to the order-filter $\{X \mid c \in X\}$ of $(\mathcal{P}(\{c_1, c_2, c_3\}), \subseteq)$. This allows us to reformulate the problem as the problem of checking the satisfiability of a family of set constraints, namely to the problem of checking whether the following family of set constraints is satisfiable:

- I_y is an upwards closed subset of $\mathcal{P}(\{c_1, c_2, c_3\})$ with respect to $\subseteq$;
- $I_{c_1 \wedge y} = I_{c_1} \cap I_y$; $\quad I_{c_2 \wedge c_3} = I_{c_2} \cap I_{c_3}$; $\quad I_{c_1 \vee y} = I_{c_1} \cup I_y$; $\quad I_{c_2 \vee c_3} = I_{c_2} \cup I_{c_3}$;
- $X \in I_{c_i}$ if and only if $c_i \in X$ for $i = 1, 2, 3$, $\quad$ for every $X \subseteq \{c_1, c_2, c_3\}$;
- $I_{c_1 \wedge y} = I_{c_2 \wedge c_3}$; $\quad I_{c_1 \vee y} = I_{c_2 \vee c_3}$.

The terms with which we associated sets are exactly the subterms of terms occurring in ϕ. We will denote by $ST(\phi)$ the set of all subterms occurring in ϕ.

By encoding every set I_e, $e \in ST(\phi)$, by a unary predicate P_e we can reduce the problem of testing the satisfiability of the family of set constraints above to the problem of testing the satisfiability of the following conjunction in first-order logic:

$$
\begin{array}{llrcll}
\text{(Her)} & & P_y(X_1) & \rightarrow & P_y(X_2) & \text{for all } X_1 \subseteq X_2 \subseteq \{c_1, c_2, c_3\} \\
\text{(Ren)} & (\wedge) & P_{c_1 \wedge y}(X) & \leftrightarrow & P_{c_1}(X) \wedge P_y(X) & \text{for all } X \subseteq \{c_1, c_2, c_3\} \\
 & & P_{c_2 \wedge c_3}(X) & \leftrightarrow & P_{c_2}(X) \wedge P_{c_3}(X) & \text{for all } X \subseteq \{c_1, c_2, c_3\} \\
 & (\vee) & P_{c_1 \vee y}(X) & \leftrightarrow & P_{c_1}(X) \vee P_y(X) & \text{for all } X \subseteq \{c_1, c_2, c_3\} \\
 & & P_{c_2 \vee c_3}(X) & \leftrightarrow & P_{c_2}(X) \vee P_{c_3}(X) & \text{for all } X \subseteq \{c_1, c_2, c_3\} \\
 & (\mathsf{c_i}) & P_{c_i}(X) & & & \text{for all } X \subseteq \{c_1, c_2, c_3\} \text{ with } c_i \in X, i = 1, 2, 3 \\
 & & \neg P_{c_i}(X) & & & \text{for all } X \subseteq \{c_1, c_2, c_3\} \text{ with } c_i \notin X, i = 1, 2, 3 \\
\text{(P)} & & P_{c_1 \wedge y}(X) & \leftrightarrow & P_{c_2 \wedge c_3}(X), & \text{for all } X \subseteq \{c_1, c_2, c_3\} \\
 & & P_{c_1 \vee y}(X) & \leftrightarrow & P_{c_2 \vee c_3}(X), & \text{for all } X \subseteq \{c_1, c_2, c_3\}.
\end{array}
$$

We obtain a structure-preserving translation to first-order logic, and, ultimately, to clause form. We can regard this clause set as a set of first-order clauses: in this case its satisfiability can be checked for instance by resolution (or by resolution with constraint clauses [Sofronie-Stokkermans 2000]). We can also consider the (finite) set of all instances of this set of clauses obtained by replacing the variables with subsets of C; then the satisfiability can be checked with any SAT-checker for propositional logic.

1.2 Structure of the paper

The paper is structured as follows. Section 2 contains the background information needed in the paper. Section 3 contains generalities on unification for bounded distributive lattices. Section 4 contains the main result of the paper: a method for deciding unification for bounded distributive lattices, and extensions to unification with linear constant restrictions and to the positive theory of D_{01}. In Section 5 we discuss the complexity of unification over (i.e. in an algebraic extension of) the free distributive lattice with n generators. In Section 6 several ways of implementing the method described in Sections 4 and 5 are presented and tested on various examples. Finally, our method is compared with other approaches. Section 7 contains conclusions and plans for future work.

It seems that many of the results in this paper can be extended to other varieties in which the free algebras have a description similar to those in D_{01} such as, for instance, the variety of De Morgan algebras, and other subvarieties of the variety of Ockham algebras[3]. However, in this paper we restrict our attention to the class of bounded distributive lattices only.

2. PRELIMINARIES

2.1 Algebra

Let Σ be a signature and $a : \Sigma \rightarrow \mathbb{N}$ an arity function. A Σ-algebra is a structure $\mathbf{A} = (A, \{\sigma_A\}_{\sigma \in \Sigma})$, where A is a non-empty set and for every $\sigma \in \Sigma$, $\sigma_A : A^{a(\sigma)} \rightarrow A$. We denote by $T_\Sigma(X)$ the term algebra over Σ in the variables X.

An *equation* is an expression of the form $t_1 = t_2$ where $t_1, t_2 \in T_\Sigma(X)$. A Σ-algebra $\mathbf{A} = (A, \{\sigma_A\}_{\sigma \in \Sigma})$ *satisfies* an equation $t_1 = t_2$ (notation: $\mathbf{A} \models t_1 = t_2$) if t_1 and t_2 become equal for every substitution of elements in A for the variables. An equational class is the class of all algebras that satisfy a set of equations. If E is a set of equations in the signature Σ, then we denote by $F_\Sigma^E(X) := T_\Sigma(X)/{\equiv_E}$ the free algebra over X in the equational class of all algebras that satisfy E (where $\equiv_E$ is the Σ-congruence on $T_\Sigma(X)$ generated by E). The equational theory of a class of structures is the set of universal atomic formulas that hold in all members of the class. For a class of algebras, this is simply the collection of all equations that hold in all members of the class.

A *system of equations* is a finite set of equations $\mathcal{S} : \{s_1 = t_1, \ldots, s_k = t_k\}$, where $s_i, t_i \in T_\Sigma(X)$ for every $1 \leq i \leq k$.

Definition 2.1. Let $\{y_1, \ldots, y_n\} \subseteq X$ be the set of all variables in $\mathcal{S}$. An algebra $\mathbf{A} = (A, \{\sigma_A\}_{\sigma \in \Sigma})$ satisfies the existential closure, $\exists y_1, \ldots y_n (s_1 = t_1 \wedge \cdots \wedge s_k = t_k)$, of $\mathcal{S}$ (notation: $\mathbf{A} \models \exists y_1, \ldots y_n (s_1 = t_1 \wedge \cdots \wedge s_k = t_k)$) if there exists a map $h : X \rightarrow A$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h} : T_\Sigma(X) \rightarrow \mathbf{A}$ is the unique homomorphism of Σ-algebras that extends h.

Definition 2.2. Let E be an equational theory, $\mathbf{A} = (A, \{\sigma_A\}_{\sigma \in \Sigma})$ a Σ-algebra, and $\mathcal{S} : \{s_1 = t_1, \ldots, s_k = t_k\}$, where $s_i, t_i \in T_\Sigma(X)$ a system of equations.

(1) $\mathcal{S}$ is *solvable in* $\mathbf{A}$ if there exists a variable assignment $h : X \rightarrow A$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h}$ is the unique homomorphism of Σ-algebras that extends h to terms.

(2) $\mathcal{S}$ is *solvable in E over* $\mathbf{A}$ if there exists an algebra $\mathbf{B}$ which satisfies all equations in E and an embedding $\mathbf{A} \hookrightarrow \mathbf{B}$, such that $\mathcal{S}$ is solvable in $\mathbf{B}$.

It is easy to see that $\mathcal{S}$ is *solvable in* $\mathbf{A}$ if and only if $\mathbf{A}$ satisfies the existential closure of $\mathcal{S}$. Satisfaction of conjunctions of equations preceded by arbitrary quantifier sequences can be defined similarily.

[3]Meanwhile, extensions to more general classes of algebras were also studied in [Sofronie-Stokkermans 2004].

2.2 E-Unification

We present the definitions and results on E-unification needed in the paper.

Definition 2.3. Let E be an equational theory, Σ its signature, and Δ a signature containing Σ. Let $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ be a system of equations, where $s_i, t_i \in T_\Delta(Y)$. Then $\mathcal{S}$ defines an E*-unification problem* over Δ.

- $\mathcal{S}$ is *elementary* if and only if $\Delta \subseteq \Sigma$;
- $\mathcal{S}$ is an E-unification problem with *(free) constants* if and only if $\Delta \backslash \Sigma$ is a set of constant symbols; and
- $\mathcal{S}$ is an E-unification problem with *linear constant restrictions* if and only if it is an E-unification problem with constants and, in addition, a linear ordering $<$ on the variables and free constants occurring in $\mathcal{S}$ is given.
- In a *general* E*-unification problem* $\Delta \backslash \Sigma$ may contain arbitrary function symbols.

Definition 2.4. A unification problem $\mathcal{S}$ *has a solution* with respect to E if there is a substitution $\sigma : Y \to T_\Delta(Y)$ such that $\sigma(s_i) \equiv_E \sigma(t_i)$ for every $1 \leq i \leq k$.

If $\mathcal{S}$ is an E-unification problem with linear constant restrictions, a solution for $\mathcal{S}$ is a substitution $\sigma : Y \to T_\Delta(Y)$ with the additional property that for every variable $y \in Y$ and every constant c, if $y < c$ then c does not occur in $\sigma(y)$.

In this context, one can study decidability of unifiability, the existence of unifiers, their classification according to "generality", or the possibility of determining minimal sets of unifiers which are complete, in the sense that all other unifiers are less general. In this paper we focus on testing unifiability. This is sufficient in many applications (e.g. in constraint-based approaches to automated deduction [Bürckert 1991; Nieuwenhuis and Rubio 1992; Kirchner and Kirchner 1989]) and is often simpler than computing complete sets of unifiers.

The unification problem with respect to an equational theory can alternatively be formulated as satisfiability in the free algebra freely generated by the constants (with additional restrictions, in the presence of linear constant restrictions). We do not present these results in their full generality, but in Section 3 we illustrate them for the case of unification with respect to D_{01}.

The importance of E-unification with linear constant restrictions is justified by its link with the positive theory of E. The *positive theory of* E is the collection of those closed formulae valid in the class of all models of E which are (equivalent to a formula) of the form

$$\phi = Q_1 x_1 \dots Q_m x_m (\bigvee_{i=1}^{q} (s_{i1} = t_{i1} \wedge \dots \wedge s_{in_i} = t_{in_i})), \tag{1}$$

where $Q_1, \dots, Q_m \in \{\exists, \forall\}$, and $x_1, \dots, x_m$ are distinct variables.

With every problem of the form (1) we can associate a set of E-unification problems with linear constant restrictions, namely the set consisting of q problems, $\mathcal{S}_1, \dots, \mathcal{S}_q$, where $\mathcal{S}_i = \{s_{i1} = t_{i1}, \dots, s_{in_i} = t_{in_i}\}$ in which the existentially quantified variables are treated as variables, the universally quantified variables are treated as constants, and the linear order $x_1 < x_2 < \cdots < x_m$ is induced by the quantifier prefix of ϕ.

PROPOSITION 2.5 [BAADER AND SNYDER 2001; BAADER AND SCHULZ 1996]. *Let $\phi = Q_1x_1 \dots Q_mx_m(\bigvee_{i=1}^{q}(s_{i1} = t_{i1} \wedge \cdots \wedge s_{in_i} = t_{in_i}))$ be a positive Σ-sentence (in disjunctive normal form). Then ϕ is valid in all models of E if and only if one of the E-unification problems with linear constant restrictions $\mathcal{S}_1, \dots, \mathcal{S}_q$ associated with ϕ as explained above has a solution.*

THEOREM 2.6 [BAADER AND SNYDER 2001; BAADER AND SCHULZ 1996]. *Let E be a non-trivial equational theory. The following statements are equivalent:*

(1) The positive theory of E is decidable.

(2) The positive theory of $F_\Sigma^E(X)$ is decidable.

(3) General E-unification is decidable.

(4) E-unification with linear constant restrictions is decidable.

2.3 Partially ordered sets and lattices

A partially ordered set is a structure $(X, \leq)$, where $\leq$ is a reflexive, antisymmetric and transitive binary relation on X.

Let $(X, \leq)$ be a partially-ordered set. An order filter of $(X, \leq)$ is a subset of X which is upwards closed with respect to $\leq$. An order ideal of $(X, \leq)$ is a subset of X which is downwards closed with respect to $\leq$. If $x \in X$, we use the notation $\uparrow x$ for the order filter $\{y \in X \mid y \geq x\}$ and the notation $\downarrow x$ for the order ideal $\{y \in X \mid y \leq x\}$. If $\mathbf{X} = (X, \leq)$ is a partially-ordered set, we denote its set of order filters by $\mathcal{O}(\mathbf{X})$. It is easy to see that there is an order isomorphism between $(\mathcal{O}(\mathbf{X}), \subseteq)$ and the set of all order-preserving maps from X to the partially-ordered set $\underline{\mathbf{2}} = (\{0, 1\}, \leq)$, where $0 < 1$, with the pointwise ordering: (i) the characteristic function of every order filter of X is an order-preserving map from X to $\underline{\mathbf{2}}$, and (ii) for every order-preserving map $f : X \to \{0, 1\}$, $f^{-1}(1) \subseteq X$ is an order filter.

A structure $\mathbf{L} = (L, \vee, \wedge)$, where L is a non-empty set and $\vee$ and $\wedge$ are two binary operations on L, is a *lattice* if $\vee$ and $\wedge$ are associative, commutative and idempotent and satisfy the absorption laws. A *distributive lattice* is a lattice that satisfies either of the distributive laws. A lattice $\mathbf{L} = (L, \vee, \wedge)$ has a *first element* if there is an element $0 \in L$ such that $0 \leq x$ for every $x \in L$; it has a *last element* if there is an element $1 \in L$ such that $x \leq 1$ for every $x \in L$ (where $x \leq y$ if and only if $x \vee y = y$). A lattice having both a first and a last element is called *bounded*.

In what follows, when we refer to bounded distributive lattices, the first and last element are supposed to be included in the signature. Thus, a bounded distributive lattice is a structure $\mathbf{L} = (L, \vee, \wedge, 0, 1)$, where $(L, \vee, \wedge)$ is a distributive lattice and 0, 1 are constants such that 0 is first element and 1 last element in $(L, \vee, \wedge)$. We denote the equational class of all bounded distributive lattices by D_{01}. D_{01} contains e.g. the two-element bounded lattice, $\mathbf{2} = (\{0, 1\}, \vee, \wedge, 0, 1)$, where $0 \vee 1 = 1, 0 \wedge 1 = 0$.

For further details we refer to [Davey and Priestley 1992].

2.4 Representation of distributive lattices

In 1933, Birkhoff [Birkhoff 1933] proved that every finite distributive lattice $\mathbf{L}$ is isomorphic to the set of all downwards closed subsets of $\mathcal{J}(\mathbf{L})$, where $\mathcal{J}(\mathbf{L})$ is the set of all join-irreducible elements of $\mathbf{L}$; the isomorphism is defined by $\eta(a) = \{x \in \mathcal{J}(\mathbf{L}), x \leq a\}$ (cf. e.g. [Davey and Priestley 1992], Theorem 8.17). This result is also known as the Birkhoff representation theorem for distributive lattices.

In what follows, the partially-ordered set $\mathbf{D}(\mathbf{L}) := (D(\mathbf{L}), \leq) = (\mathcal{J}(\mathbf{L}), \geq)$, obtained by reversing the order between the elements of $\mathcal{J}(\mathbf{L})$, will be called the (Birkhoff) dual of $\mathbf{L}$.

The dual of a finite distributive lattice is much smaller and less complex than the lattice itself. Therefore, problems concerning finite distributive lattices are likely to become simpler when translated into problems about their duals. We illustrate this by comparing the free algebra in D_{01} over a finite set C, $F_{\mathsf{D}_{01}}(C)$, and its Birkhoff dual $\mathbf{D}(F_{\mathsf{D}_{01}}(C))$.

COROLLARY 2.7. *Let C be a finite set. The following statements hold:*

(1) The map $p_C : (D(F_{\mathsf{D}_{01}}(C)), \leq) \to (\mathcal{P}(C), \subseteq)$ defined for every $j \in D(F_{\mathsf{D}_{01}}(C))$ by $p_C(j) = \{c \in C \mid j \leq c\}$ is an order isomorphism.

(2) The map $\eta_C : F_{\mathsf{D}_{01}}(C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ defined for every $t \in F_{\mathsf{D}_{01}}(C)$ by $\eta_C(t) = \{X \subseteq C \mid \bigwedge_{c \in X} c \leq t\}$ is a lattice isomorphism.
Its inverse is defined for every $U \in \mathcal{O}(\mathcal{P}(C), \subseteq)$ by $\eta_C^{-1}(U) = \bigvee_{X \in U}(\bigwedge_{c \in X} c)$.

Proof: (1) An element j of $F_{\mathsf{D}_{01}}(C)$ is join-irreducible if and only if it can be written as a conjunction of the generators, i.e. if and only if there exists $X \subseteq C$ such that $j = \bigwedge_{c \in X} c$. It is easy to see that in this case $p_C(j) = X$. p_C is obviously a bijection; it converse is the map which associates with every $X \subseteq C$ the (join-irreducible) element $\bigwedge_{c \in X} c$ of $D(F_{\mathsf{D}_{01}}(C))$. Clearly, if $j_1 \geq j_2$ then $p_C(j_1) \subseteq p_C(j_2)$, and if $X_1 \subseteq X_2$ then $\bigwedge_{c \in X_1} c \geq \bigwedge_{d \in X_2} d$. Hence, p_C is an order isomorphism.

(2) By Birkhoff's representation theorem we know that there is an isomorphism from $F_{\mathsf{D}_{01}}(C)$ to the set of order-ideals of $\mathcal{J}(F_{\mathsf{D}_{01}}(C))$, defined by $\eta(t) = \{j \in \mathcal{J}(F_{\mathsf{D}_{01}}(C)) \mid j \leq t\}$ for every $t \in F_{\mathsf{D}_{01}}(C)$; its inverse is defined for every order ideal U of $\mathcal{J}(F_{\mathsf{D}_{01}}(C))$ by $\eta^{-1}(U) = \bigvee_{j \in U} j$. By (1), the set of order ideals of $\mathcal{J}(\mathbf{L})$ is order-isomorphic with the set of order-filters of $(\mathcal{P}(C), \subseteq)$. It is easy to see that the composition of these isomorphisms yields the lattice isomorphism $\eta_C : F_{\mathsf{D}_{01}}(C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ defined by $\eta_C(t) = \{X \subseteq C \mid \bigwedge_{c \in X} c \leq t\}$; its inverse is defined for every $U \in \mathcal{O}(\mathcal{P}(C), \subseteq)$ by $\eta_C^{-1}(U) = \bigvee_{X \in U}(\bigwedge_{c \in X} c)$. □

Every member of $F_{\mathsf{D}_{01}}(C)$ can be written as a finite join of finite meets of elements in C. Hence, $F_{\mathsf{D}_{01}}(C)$ is finite, and its number of elements is bounded by $2^{2^{|C|}}$. $|F_{\mathsf{D}_{01}}(C)|$ has been computed only for small values of $|C|$. By Corollary 2.7(1), $(D(F_{\mathsf{D}_{01}}(C)), \leq)$ is order isomorphic to $(\mathcal{P}(C), \subseteq)$, hence has $2^{|C|}$ elements.

The main idea of this paper relies on this remark. The relatively simple structure of $D(F_{\mathsf{D}_{01}}(C))$ allows us to define a more efficient method for checking the satisfiability of unification problems with constants compared with methods that use the structure of $F_{\mathsf{D}_{01}}(C)$ and/or equational reasoning.

3. UNIFICATION IN D_{01}: GENERALITIES

In this section we present some general properties of unification for D_{01}.

We start by showing that unification with respect to D_{01} can alternatively be formulated as a satisfiability problem in the free bounded distributive lattice freely generated by the constants, with additional restrictions, in the presence of linear constant restrictions. We use this result for proving that unification in D_{01} is decidable.

LEMMA 3.1. *Let $\mathcal{S} = \{s_1 = t_1, \dots, s_n = t_n\}$ be a D_{01}-unification problem with constants C, variables Y and linear constant restrictions lcr. Then the following are equivalent:*

(1) $\mathcal{S}$ has a solution with respect to D_{01}.

(2) There exists $h : Y \to F_{\mathsf{D}_{01}}(C)$ with the following properties:
- *(i) $\overline{h}(s_i) = \overline{h}(t_i)$ for all $i \in \{1, \dots, n\}$ (where $\overline{h} : T_{\mathsf{D}_{01}}(Y \cup C) \to F_{D_{01}}(C)$ is the unique extension of h to a homomorphism, such that, for all $c \in C$, $\overline{h}(c) = [c]$, the equivalence class of c in $F_{D_{01}}(C)$), and,*
- *(ii) for every variable $y \in Y$ and every constant c, if $y < c$ is in lcr then there exists a term $t^y_c \in T_{\mathsf{D}_{01}}(C\backslash\{c\})$ such that $h(y) = [t^y_c]$.*

Proof: (1) $\Rightarrow$ (2) From every solution $\sigma : Y \to T_{\mathsf{D}_{01} \cup C}(Y)$ of $\mathcal{S}$ we can construct a map $h : Y \to F_{\mathsf{D}_{01}}(C)$, defined by $h(y) := \pi(g(\sigma(y)))$, where $g : T_{\mathsf{D}_{01}\cup C}(Y) \to T_{\mathsf{D}_{01}\cup C}(\emptyset) = T_{\mathsf{D}_{01}}(C)$ is the canonical extension of the map that maps every element of Y to 0, and $\pi : T_{\mathsf{D}_{01}}(C) \to F_{\mathsf{D}_{01}}(C)$ is the canonical surjection associated with the congruence $\equiv_{\mathsf{D}_{01}}$. As $\overline{\sigma}(s_i) \equiv_{\mathsf{D}_{01}} \overline{\sigma}(t_i)$ for all $i \in \{1, \dots, n\}$ we know that for every $L \in D_{01}$ and every $f : Y \cup C \to L$, $\overline{f}(\overline{\sigma}(s_i)) = \overline{f}(\overline{\sigma}(t_i))$ for all $i \in \{1, \dots, n\}$. Hence, in particular, $\overline{h}(s_i) = \overline{\pi \circ g}(\overline{\sigma}(s_i)) = \overline{\pi \circ g}(\overline{\sigma}(t_i)) = \overline{h}(t_i)$ for all $i \in \{1, \dots, n\}$. Moreover, if $y \in Y$ and $c \in C$ and $y < c$ then, by assumption $\sigma(y)$ does not contain c, hence (2)(ii) holds for $t^y_c = \sigma(y)$.

(2) $\Rightarrow$ (1) Assume that (2) holds. Note first that if $y < c_1$ and $y < c_2$ then there exists a term t^y which neither contains c_1 nor c_2 such that $h(y) = [t^y]$. Indeed, as $y < c_i, i = 1, 2$, there exist terms $t^y_{c_i}$, $i = 1, 2$, not containing c_i, such that $h(y) = [t^y_{c_i}]$. Hence $t^y_{c_1} \equiv_{\mathsf{D}_{01}} t^y_{c_2}$. Let $\tau : Y \cup C \to T_{\mathsf{D}_{01}\cup C}(Y)$ be the substitution that replaces c_1 with 0 and leaves the other variables unchanged. Then $\tau(t^y_{c_1}) \equiv_{\mathsf{D}_{01}} \tau(t^y_{c_2})$. As $t^y_{c_1}$ does not contain c_1, $\tau(t^y_{c_1}) = t^y_{c_1}$. On the other hand, as $t^y_{c_2}$ does not contain c_2, $\tau(t^y_{c_2})$ contains neither c_1 nor c_2. This shows that there exists a term $t^y := \tau(t^y_{c_2})$ which contains neither c_1 nor c_2, such that $h(y) = [t^y_{c_1}] = [t^y_{c_2}] = [t^y]$. The argument can easily be extended to show that if $y < c_1, \dots, y < c_n$ are in lcr then there exists a term t^y which does not contain $c_1, \dots, c_n$ such that $h(y) = [t^y]$.

Define $\sigma : Y \to T_{\mathsf{D}_{01}}(C)$ by $\sigma(y) = t$, an arbitrary representative of the equivalence class of $h(y)$, if y occurs in no linear constant restriction; otherwise, if y occurs in the constant restrictions $y < c_1, \dots y < c_n$, $\sigma(y) = t^y$, where t^y is a term which does not contain $c_1, \dots, c_n$, such that $h(y) = [t^y]$, which exists by the remark above. This definition ensures that if $y < c$ in the linear constant restrictions then c does not occur in $\sigma(y)$. It is easy to check that for every term $t \in T_{\mathsf{D}_{01}}(C)$,

$\overline{h}(t) = [\overline{\sigma}(t)]$. Hence, for every $s_i = t_i \in \mathcal{S}$ we have $[\overline{\sigma}(s_i)] = \overline{h}(s_i) = \overline{h}(t_i) = [\overline{\sigma}(t_i)]$, so $\overline{\sigma}(s_i) \equiv_{\mathsf{D}_{01}} \overline{\sigma}(t_i)$. □

As $F_{\mathsf{D}_{01}}(C)$ is finite for every finite C, it follows that D_{01}-unification with free constants, and D_{01}-unification with linear constant restrictions are decidable. D_{01}-unification with free constants is co-NP hard: if $\mathcal{S}$ contains only one equation and no variables it reduces to the word problem for D_{01}, which has been shown to be co-NP hard [Hunt et al. 1987].

3.1 D_{01}-unification without constants

By Lemma 3.1, a D_{01}-unification problem $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ with variables $\{y_1, \dots, y_n\}$ and no constants has a solution if and only if the formula $\exists y_1, \dots y_n(s_1 = t_1 \wedge \dots \wedge s_k = t_k)$ is valid in $F_{\mathsf{D}_{01}}(\emptyset)$, which is the two-element lattice.

THEOREM 3.2. *Let $\mathcal{S}$ be a D_{01}-unification problem without constants. Assume that all terms in $\mathcal{S}$ have been simplified by (recursively) applying the following simplification rules*[4]*: $e \wedge 1 \mapsto e; e \wedge 0 \mapsto 0; e \vee 1 \mapsto 1; e \vee 0 \mapsto e$.*

(1) If $\{0, 1\} \not\subseteq ST(\mathcal{S})$, then $\mathcal{S}$ always has a solution.

(2) If $\{0, 1\} \subseteq ST(\mathcal{S})$ and $\mathcal{S}$ consists of only one equation (or else it contains the equation $1 = 0$) then $\mathcal{S}$ has no solution.

(3) If $\mathcal{S}$ only contains the operators $\wedge, 0, 1$, then the problem of checking whether $\mathcal{S}$ has a solution can be solved in polynomial time. The same holds if $\mathcal{S}$ only contains the operators $\vee, 0, 1$.

(4) In general, the problem of checking whether $\mathcal{S}$ has a solution is NP-complete.

Proof: (1) Assume that $1 \notin ST(\mathcal{S})$. Then the substitution that assigns 0 to every variable occurring in $\mathcal{S}$ is a solution for $\mathcal{S}$. The case $0 \notin ST(\mathcal{S})$ can be proved similarly. (2) is obvious if all terms in $\mathcal{S}$ are simplified, because normalized terms containing 0 resp. 1 are equal to 0 resp. 1.

(3) Assume that $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ only contains the operators $\wedge, 0, 1$. $\mathcal{S}$ has a solution with respect to the equational theory of D_{01} if and only if its existential closure is valid in the meet semilattice with two elements, $S_2 = (\{0, 1\}, \mathsf{min}, 0, 1)$. It is easy to see that $S_2 \models \exists y_1 \dots y_n(s_1 = t_1 \wedge \dots \wedge s_k = t_k)$ if and only if $S_2 \not\models \forall y_1 \dots y_n[(s_1 = t_1 \wedge \dots \wedge s_k = t_k) \rightarrow 0 = 1]$. Since the uniform word problem in semilattices is polynomial time decidable[5], it follows that we can decide in polynomial time whether $\mathcal{S}$ has a solution.

(4) The problem of deciding whether a unification problem without constants has a solution is clearly in NP, because it is a special instance of the Boolean satisfiability problem. NP-hardness follows from the fact that the satisfiability problem

[4]This can be done in polynomial time.

[5]The polynomial time decidability of the uniform word problem for semilattices is a consequence of the fact that every poset embeds into a semilattice (in fact into a lattice). It is known (cf. [Burris 1995]) that if every partial algebra model of an equational theory K (weakly) embeds into a total model of K, then the uniform word problem for K is decidable in PTIME. (This argument was first used by Skolem in 1920 in the study of the uniform word problem in lattices.) The polynomial time decidability of the uniform word problem for semilattices also follows from results of Narendran on unification modulo ACI + 1 + 0 [Narendran 1996].

for Boolean formulae of the form $E = F \wedge \neg G$, where F and G only contain the operators $\vee$ and $\wedge$ (which is NP-complete [Hunt and Stearns 1990]), can be reduced in polynomial time to the satisfiability of a D_{01}-unification problem. Indeed let $E = F \wedge \neg G$, where F and G only contain the operators $\vee$ and $\wedge$. Starting from E we can construct the unification problem $\mathcal{S} : \{F = 1, G = 0\}$. It is easy to see that $\mathcal{S}$ has a solution if and only if E is satisfiable. □

4. UNIFICATION IN D_{01}

This section contains the main contribution of this paper, namely an algorithm for deciding whether a D_{01}-unification problem with free constants (or, more generally, with linear constant restrictions) has a solution. We show that checking whether a unification problem with free constants (resp. with linear constant restrictions) has a solution with respect to D_{01} can be reduced to the problem of deciding the satisfiability of a set of first-order formulae in classical logic without equality.

In what follows we implicitly assume that the unification problems we consider contain at least one existentially quantified variable. If there are no existentially quantified variables the unification problem reduces to the word problem for D_{01} which is known to be co-NP complete [Hunt et al. 1987].

4.1 Unification with free constants in D_{01}

By Lemma 3.1, specialized to the case when the set of linear constant restrictions is empty, if $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ is a D_{01}-unification problem with free constants in C and variables Y then $\mathcal{S}$ has a solution if and only if there exists $h : Y \to F_{\mathsf{D}_{01}}(C)$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h} : T_{\underline{\mathsf{D}_{01}}}(Y \cup C) \to F_{\mathsf{D}_{01}}(C)$ is the unique extension of h to a homomorphism, such that $\overline{h}(c) = [c]$ for all $c \in C$.

We now reduce the problem of testing the satisfiability of a unification problem $\mathcal{S}$ to that of checking the satisfiability of a set of clauses. We do this in two steps: Theorem 4.1 shows that $F_{\mathsf{D}_{01}}(C)$ can be replaced with the lattice of order filters of $(\mathcal{P}(C), \subseteq)$; Theorem 4.2 further reduces the problem to that of checking the satisfiability of a set of (ground) clauses.

THEOREM 4.1. *For any D_{01}-unification problem $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ with free constants C and variables $Y = \{y_1, \dots, y_n\}$, the following are equivalent:*

(1) $\mathcal{S}$ has a solution with respect to D_{01}.

(2) There exists $h : Y \to F_{\mathsf{D}_{01}}(C)$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h} : T_\Sigma(Y \cup C) \to F_{\mathsf{D}_{01}}(C)$ is the unique homomorphism that extends h, such that $\overline{h}(c) = [c]$ for all $c \in C$.

(3) There exists $g : Y \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ such that $\overline{g}(s_i) = \overline{g}(t_i)$ for every $1 \leq i \leq k$, where $\overline{g} : T_\Sigma(Y \cup C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ is the unique homomorphism that extends g, such that $\overline{g}(c) = \uparrow\{c\} = \{X \subseteq C \mid c \in X\}$ for every $c \in C$.

Proof: The equivalence of (1) and (2) follows directly from Lemma 3.1.

(2) $\Rightarrow$ (3) Let $h : \{y_1, \dots, y_n\} \to F_{\mathsf{D}_{01}}(C)$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h}$ is the homomorphic extension of h to $T_\Sigma(Y \cup C)$, with $\overline{h}(c) = [c]$

for all $c \in C$. Define $g := \eta_C \circ h : \{y_1, \dots, y_n\} \to \mathcal{O}(\mathcal{P}(C), \subseteq)$. Let $\overline{g} : T_\Sigma(Y \cup C) \to F_{\mathsf{D}_{01}}(C)$ be the (unique) homomorphic extension of g, with $\overline{g}(c) = \uparrow\{c\}$. As $\eta_C(\overline{h}(c)) = \eta_C([c]) = \uparrow\{c\}$, it follows that $\overline{g} = \eta_C \circ \overline{h}$. Then, for every $1 \leq i \leq k$, $\overline{g}(s_i) = \eta_C(\overline{h}(s_i)) = \eta_C(\overline{h}(t_i)) = \overline{g}(t_i)$.

(3) $\Rightarrow$ (2) Let $g : \{y_1, \dots, y_n\} \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ such that $\overline{g}(s_i) = \overline{g}(t_i)$ for every $1 \leq i \leq k$, where $\overline{g} : T_\Sigma(Y \cup C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ is the homomorphic extension of g, such that $\overline{g}(c) = \uparrow\{c\}$ for every $c \in C$. Then $h : \{y_1, \dots, y_n\} \to F_{\mathsf{D}_{01}}(C)$ can be defined by $h := \eta_C^{-1} \circ g$. It can be checked that if $\overline{h} : T_\Sigma(Y \cup C) \to F_{\mathsf{D}_{01}}(C)$ is the unique homomorphism that extends h with $\overline{h}(c) = [c]$ for every $c \in C$, then $\overline{h} = \eta_C^{-1} \circ \overline{g}$, hence for every $1 \leq i \leq k$, $\overline{h}(s_i) = \eta_C^{-1}(\overline{g}(s_i)) = \eta_C^{-1}(\overline{g}(t_i)) = \overline{h}(t_i)$. $\square$

Theorem 4.1 justifies a reduction of the problem of checking whether a unification problem with constants $\mathcal{S}$ has a solution to the problem of checking the satisfiability of a system of set constraints. This reduction can be used to give a structure-preserving translation to clause form. Thus, the problem of checking whether a unification problem with constants $\mathcal{S}$ has a solution can be reduced to the problem of checking the satisfiability of a set of clauses.

THEOREM 4.2. *Let $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ be a D_{01}-unification problem with free constants C, and variables $Y = \{y_1, \dots, y_n\}$. Let $ST(\mathcal{S})$ be the set of all subterms of terms occurring in $\mathcal{S}$. The following are equivalent:*

(1) There exists $h : \{y_1, \dots, y_n\} \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h} : T_\Sigma(Y \cup C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ is the unique homomorphism that extends h, such that $\overline{h}(c) = \uparrow\{c\}$ for every $c \in C$.

(2) There exists a family $\{I_e\}_{e \in ST(\mathcal{S})}$, such that $I_e \subseteq \mathcal{P}(C)$ for all $e \in ST(\mathcal{S})$, and for all $X, X_1, X_2 \subseteq C$ the following hold:
- *if $X_1 \in I_y$ and $X_1 \subseteq X_2$ then $X_2 \in I_y$, for every $y \in \{y_1, \dots, y_n\}$;*
- *$I_{e_1 \wedge e_2} = I_{e_1} \cap I_{e_2}$; $I_{e_1 \vee e_2} = I_{e_1} \cup I_{e_2}$;*
- *$I_0 = \emptyset$; $I_1 = \mathcal{P}(C)$;   and for every $c \in C$, $X \in I_c$ if and only if $c \in X$;*
- *$I_{s_i} = I_{t_i}$ for all $1 \leq i \leq k$.*

(3) The conjunction of the following formulae (in a signature which contains unary predicate symbols $\{P_e \mid e \in ST(\mathcal{S})\}$, a constant D for every subset D of C, and no other function symbols) is satisfiable:

(Her)		$P_y(X_1) \to P_y(X_2)$	*for all* $X_1 \subseteq X_2 \subseteq C, y \in \{y_1, \dots, y_n\}$
(Ren)	($\wedge$n)	$P_{e_1 \wedge e_2}(X) \to P_{e_i}(X)$	*for all* $X \subseteq C, i = 1, 2$
	($\wedge$p)	$P_{e_1}(X) \wedge P_{e_2}(X) \to P_{e_1 \wedge e_2}(X)$	*for all* $X \subseteq C$
	($\vee$n)	$P_{e_1 \vee e_2}(X) \to P_{e_1}(X) \vee P_{e_2}(X)$	*for all* $X \subseteq C$
	($\vee$p)	$P_{e_i}(X) \to P_{e_1 \vee e_2}(X)$	*for all* $X \subseteq C, i = 1, 2$
	(1)	$P_1(X)$	*for all* $X \subseteq C$
	(0)	$\neg P_0(X)$	*for all* $X \subseteq C$
	(cp)	$P_c(X)$	*for all* $X \subseteq C$ *with* $c \in X$
	(cn)	$\neg P_c(X)$	*for all* $X \subseteq C$ *with* $c \notin X$
(P)		$P_{s_i}(X) \leftrightarrow P_{t_i}(X)$,	*for all* $X \subseteq C$, *for all* $1 \leq i \leq k$

where each formula in (Her) $\cup$ (Ren) $\cup$ (P) *is the conjunction of all formulae obtained by instantiating the variables X, resp. X_1, X_2 with subsets of C (which can be considered to be constants) satisfying the additional conditions; the indices*

$e_1 \vee e_2, e_1 \wedge e_2, 0, 1, c$ range over all elements in $ST(\mathcal{S})$; y ranges over all variables in $\{y_1, \dots, y_n\}$.

Proof: (1) $\Rightarrow$ (2). For every $e \in ST(\mathcal{S})$ let $I_e := \overline{h}(e)$. Since $\overline{h}$ is a 0,1-homomorphism with $\overline{h}(c) = \uparrow\{c\}$, and the lattice operations in $\mathcal{O}(\mathcal{P}(C), \subseteq)$ are union and intersection, the family $\{I_e\}_{e \in ST(\mathcal{S})}$ satisfies the conditions in (2).

(2) $\Rightarrow$ (3) Let $\{I_e\}_{e \in ST(\mathcal{S})}$ be a family satisfying the conditions in (2). Then $(\mathcal{P}(C), \mathcal{I})$, where $\mathcal{I}(P_e) := I_e$ for all $e \in ST(\mathcal{S})$, is a model for $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$.

(3) $\Rightarrow$ (1) Assume that $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ (which can be regarded as a conjunction of ground clauses) is satisfied by the map $\mathcal{I} : \{P_e(X) \mid e \in ST(\mathcal{S}), X \subseteq C\} \to \{0, 1\}$. For every $y \in Y$ let $h(y) := \{X \in \mathcal{P}(C) \mid \mathcal{I}(P_y(X)) = 1\}$. Let $\overline{h} : T_\Sigma(Y \cup C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ be the unique homomorphism that extends h, such that $\overline{h}(c) = \uparrow\{c\}$ for every $c \in C$. As $\mathcal{I}$ satisfies $(\mathsf{Her}) \cup (\mathsf{Ren})$, $\overline{h}(e) = \{X \in \mathcal{P}(C) \mid \mathcal{I}(P_e(X)) = 1\}$ for all $e \in ST(\mathcal{S})$. Since $\mathcal{I}$ satisfies (P), $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$. $\square$

COROLLARY 4.3. *The D_{01}-unification problem $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ with free constants C has a solution with respect to the equational theory of D_{01} if and only if the set of clauses $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ is satisfiable.*

The satisfiability of $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ can be checked, for instance, by resolution or by using a satisfiability checker.

As a direct consequence of Theorem 4.2, we obtain an upper bound on the complexity of the problem of deciding the unifiability of $\mathcal{S}$.

THEOREM 4.4. *(1) The problem of deciding whether the D_{01}-unification problem $\mathcal{S}$ has a solution can be solved in* NEXPTIME *(and in time exponential in $|ST(\mathcal{S})|2^{|C|}$ by using resolution).*

(2) If $\mathcal{S}$ only contains the operation symbols $\wedge, 0, 1$, and, possibly, constants, then the problem can be decided in EXPTIME. *The same holds if $\mathcal{S}$ only contains the operation symbols $\vee, 0, 1$, and, possibly, constants.*

Proof: Note first that the structure-preserving translation to clause form in Theorem 4.2, as well as the size of the conjunction of all formulae in $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$, is linear in $(2|ST(\mathcal{S})| + |C| \cdot |Y|) \cdot 2^{|C|}$.

(1) follows from this and the fact that the number of all distinct literals that can occur in the conjunction of ground clauses $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ in Theorem 4.2(3) is bounded by $|ST(\mathcal{S})|2^{|C|}$.

To prove (2) note that if only the operators $\wedge, 0, 1$ and, possibly, constants, occur in $\mathcal{S}$, then the clause form of $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ is a set of ground Horn clauses. Dowling and Gallier [Dowling and Gallier 1984] showed that satisfiability of a set Φ of ground Horn clauses can be proved in linear time with respect to the number of clauses in Φ. The situation when $\mathcal{S}$ only contains the operation symbols $\vee, 0, 1$, and constants follows from (2) by using the duality principle in lattices. $\square$

Remark 4.5. If $C = \emptyset$ the number of clauses corresponding to $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ in Theorem 4.2 is polynomial in $ST(\mathcal{S})$. This allows us to justify in a different way the complexity bounds settled in Theorem 3.2. For instance, item (1) of Theorem 3.2 can be proved as follows. Assume that $1 \notin ST(\mathcal{S})$. Let $\Phi_{\mathcal{S}}$ be the set of clauses associated to S by the structure-preserving translation to clause form in

Theorem 4.2(3). Since no constant occurs in $\mathcal{S}$, all the clauses $\Phi_{\mathcal{S}}$ are non-positive, so $\Phi_{\mathcal{S}}$ is satisfiable (consider a selection function that selects all negative literals in all clauses; then no resolution inference is possible). The case $0 \notin ST(\mathcal{S})$ follows by duality. (3) follows from the second part of Theorem 4.4.

4.2 Unification with linear constant restrictions

The solution of a D_{01}-unification problem $\mathcal{S}$ with linear constant restrictions is a solution for $\mathcal{S}$ with the additional property that for every variable y and every constant c, if $y < c$ then c does not occur in $\sigma(y)$.

By Lemma 3.1, if $\mathcal{S} = \{s_1 = t_1, \ldots, s_n = t_n\}$ is an D_{01}-unification problem with linear constant restrictions, then $\mathcal{S}$ has a solution with respect to D_{01} if and only if there exists $h : Y \to F_{\mathsf{D}_{01}}(C)$ such that (i) $\overline{h}(s_i) = \overline{h}(t_i)$ for all $i \in \{1, \ldots, n\}$ (where $\overline{h} : T_{\mathsf{D}_{01}}(Y \cup C) \to F_{D_{01}}(C)$ is the unique extension of h to a homomorphism, such that, for all $c \in C$, $\overline{h}(c) = [c]$, the equivalence class of c in $F_{D_{01}}(C)$), and, (ii) for every variable $y \in Y$ and every constant c, if $y < c$ then there exists a term $t^y_c \in T_{\mathsf{D}_{01}}(C \backslash \{c\})$ such that $h(y) = [t^y_c]$.

In what follows we give an alternative characterization of condition (ii) above. Let $\overline{\imath} : F_{\mathsf{D}_{01}}(C \backslash \{c\}) \to F_{\mathsf{D}_{01}}(C)$ be the unique homomorphism which extends the map $i : C \backslash \{c\} \to F_{\mathsf{D}_{01}}(C)$ defined by $i(d) = [d]$ for every $d \in C \backslash \{c\}$. $\overline{\imath}$ is an injective homomorphism. It is easy to see that for every $t \in F_{\mathsf{D}_{01}}(C)$, $t = [s]$ for some term $s \in T_{\mathsf{D}_{01}}(C \backslash \{c\})$ if and only if $t = \overline{\imath}(t')$ for some $t' \in F_{\mathsf{D}_{01}}(C \backslash \{c\})$. We express the latter condition by using the isomorphism $\eta_C : F_{\mathsf{D}_{01}}(C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ defined for every $t \in F_{\mathsf{D}_{01}}(C)$ by (cf. Corollary 2.7) $\eta_C(t) = \{X \subseteq C \mid \bigwedge_{c \in X} c \leq t\}$.

LEMMA 4.6. *Let $t \in F_{\mathsf{D}_{01}}(C)$. The following are equivalent:*

(1) There exists $t' \in F_{\mathsf{D}_{01}}(C \backslash \{c\})$ such that $t = \overline{\imath}(t')$.

(2) For every $X \in \eta_C(t)$, $X \backslash \{c\} \in \eta_C(t)$.

Proof: Consider the following diagram, where $j := \eta_C \circ \overline{\imath} \circ \eta^{-1}_{C \backslash \{c\}}$.

$$\begin{array}{ccc}
F_{\mathsf{D}_{01}}(C \backslash \{c\}) & \xrightarrow{\ \overline{\imath}\ } & F_{\mathsf{D}_{01}}(C) \\
\downarrow{\scriptstyle \eta_{C \backslash \{c\}}} & & \downarrow{\scriptstyle \eta_C} \\
\mathcal{O}(\mathcal{P}(C \backslash \{c\}), \subseteq) & \xrightarrow[\ j\ ]{} & \mathcal{O}(\mathcal{P}(C), \subseteq)
\end{array}$$

We show that for every $U \in \mathcal{O}(\mathcal{P}(C \backslash \{c\}), \subseteq)$, $j(U) = \{X \subseteq C \mid \exists Z \in U : Z \subseteq X\}$, the order filter of $(\mathcal{P}(C), \subseteq)$ generated by U. Let $U \in \mathcal{O}(\mathcal{P}(C \backslash \{c\}), \subseteq)$.

$$\begin{aligned}
j(U) &= \eta_C(\overline{\imath}(\eta^{-1}_{C \backslash \{c\}}(U))) = \eta_C(\bigvee_{Z \in U} \bigwedge_{d \in Z} d) = \{X \subseteq C \mid \bigwedge_{c \in X} c \leq \bigvee_{Z \in U} \bigwedge_{d \in Z} d\} = \\
&= \{X \subseteq C \mid \exists Z \in U : Z \subseteq X\}.
\end{aligned}$$

(1) $\Rightarrow$ (2) Let $t = \overline{\imath}(t')$, where $t' \in F_{\mathsf{D}_{01}}(C \backslash \{c\})$. Then $\eta_C(t) = \eta_C(\overline{\imath}(t')) = j(\eta_{C \backslash \{c\}}(t')) = \{X \subseteq C \mid \exists Z \in \eta_{C \backslash \{c\}}(t') : Z \subseteq X\}$, where $c \notin Z$ for every $Z \in \eta_{C \backslash \{c\}}(t')$. Then, for every $X \in \eta_C(t)$, there exists $Z \in \eta_{C \backslash \{c\}}(t')$ such that (i) $Z \subseteq X$ and (ii) $c \notin Z$, i.e. $Z \subseteq X \backslash \{c\}$. Hence, $\{X \backslash \{c\} \mid X \in \eta_C(t)\} \subseteq \eta_C(t)$.

(2) $\Rightarrow$ (1) It is easy to see that if (2) holds, then $\eta_C(t) = \{X \mid \exists Z \in U, Z \subseteq X\}$, where $U = \{X \backslash \{c\} \mid X \in \eta_C(t)\}$. U is an order filter of $(\mathcal{P}(C\backslash\{c\}), \subseteq)$, so by the surjectivity of $\eta_{C\backslash\{c\}}$, there exists $t' \in F_{\mathsf{D}_{01}}(C\backslash\{c\})$ such that $\eta_{C\backslash\{c\}}(t') = U$. Therefore, $\eta_C(\overline{\iota}(t')) = j(\eta_{C\backslash\{c\}}(t'))$ and $\eta_C(t)$ are both equal to $\{X \mid \exists Z \in U, Z \subseteq X\}$. By the injectivity of η_C it follows that $\overline{\iota}(t')$ and t are equal in $F_{\mathsf{D}_{01}}(C)$. □

We show that a structure preserving translation to clause form similar to that established in Theorems 4.1 and 4.2 allows us to obtain an algorithm for unification with linear constant restrictions.

THEOREM 4.7. *Let $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ be a D_{01}-unification problem with linear constant restrictions* lcr*, constants C and variables Y. The following are equivalent:*

(1) $\mathcal{S}$ has a solution with respect to the equational theory of D_{01}.

(2) There exists $h : Y \to F_{\mathsf{D}_{01}}(C)$ such that whenever $y < c$ is a linear constant restriction, there exists $t'_y \in F_{\mathsf{D}_{01}}(C\backslash\{c\})$ such that[6] *$h(y) = \overline{\iota}(t'_y)$ and $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$ (where $\overline{h}$ is the unique extension of h to $T_\Sigma(Y \cup C)$ to a $\{\vee, \wedge, 0, 1\}$-homomorphism, such that $\overline{h}(c) = [c]$ for all $c \in C$).*

(3) There exists $h : Y \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ such that (i) if $y < c$ is a linear constant restriction, then for every $X \in h(y)$ we have also $X\backslash\{c\} \in h(y)$; and (ii) $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$ (where $\overline{h} : T_\Sigma(Y \cup C) \to \mathcal{O}(\mathcal{P}(C), \subseteq)$ is the unique homomorphism that extends h, such that $\overline{h}(c) = \uparrow\{c\}$ for every $c \in C$).

Proof: The equivalence of (1) and (2) follows immediately from Lemma 3.1. The equivalence of (2) and (3) follows by combining the arguments of Theorem 4.1 with the results in Lemma 4.6. □

As in the case of unification with free constants, the remark above justifies a structure-preserving translation to clause form.

THEOREM 4.8. *Let $\mathcal{S} : \{s_1 = t_1, \dots, s_k = t_k\}$ be a D_{01}-unification problem with linear constant restrictions* lcr*, constants C and variables Y. The following are equivalent:*

(1) $\mathcal{S}$ has a solution with respect to the equational theory of D_{01}.

(2) There exists a family $\{I_e\}_{e \in ST(\mathcal{S})}$ indexed by the family $ST(\mathcal{S})$ of all subterms that occur in S such that for all $e \in ST(\mathcal{S})$, $I_e \subseteq \mathcal{P}(C)$, and for all $y \in Y$, and all $X, X_1, X_2 \subseteq C$ the following hold:

- *if $X_1 \in I_y$ and $X_1 \subseteq X_2$ then $X_2 \in I_y$;*
- *$I_{e_1 \wedge e_2} = I_{e_1} \cap I_{e_2}$; $I_{e_1 \vee e_2} = I_{e_1} \cup I_{e_2}$;*
- *$I_0 = \emptyset$; $I_1 = C$; and for every $c \in C$, $X \in I_c$ if and only if $c \in X$;*
- *if $y < c \in$ lcr if $X \in I_y$ then $X\backslash\{c\} \in I_y$;*
- *$I_{s_i} = I_{t_i}$ for all $1 \leq i \leq k$.*

[6]Recall that $\overline{\iota} : F_{\mathsf{D}_{01}}(C\backslash\{c\}) \to F_{\mathsf{D}_{01}}(C)$ is the unique homomorphism which extends the map $i : C\backslash\{c\} \to F_{\mathsf{D}_{01}}(C)$.

(3) The conjunction of the following set of formulae is satisfiable:

$$
\begin{array}{llcll}
(\mathsf{Her}) & P_y(X_1) & \rightarrow & P_y(X_2) & \textit{for all } X_1 \subseteq X_2 \subseteq C, y \in Y \\
(\mathsf{Ren}) & P_{e_1 \wedge e_2}(X) & \leftrightarrow & P_{e_1}(X) \wedge P_{e_2}(X) & \textit{for all } X \subseteq C \\
 & P_{e_1 \vee e_2}(X) & \leftrightarrow & P_{e_1}(X) \vee P_{e_2}(X) & \textit{for all } X \subseteq C \\
 & P_1(X) & & & \textit{for all } X \subseteq C \\
 & \neg P_0(X) & & & \textit{for all } X \subseteq C \\
 & P_c(X) & & & \textit{for all } X \subseteq C \textit{ with } c \in X \\
 & \neg P_c(X) & & & \textit{for all } X \subseteq C \textit{ with } c \notin X \\
(\mathsf{Lcr}) & P_y(X) & \rightarrow & P_y(X \backslash \{c\}) & \textit{for all } X \subseteq C \textit{ if } y < c \in \mathsf{lcr} \\
(\mathsf{P}) & P_{s_i}(X) & \leftrightarrow & P_{t_i}(X), & \textit{for all } 1 \leq i \leq k.
\end{array}
$$

Proof: This follows from Definition 2.4, Lemma 4.6, and arguments similar to those used to prove Theorem 4.2. □

As before, a NEXPTIME upper bound for the complexity of the problem can be easily established.

THEOREM 4.9. *(1) The problem of deciding whether the D_{01}-unification problem $\mathcal{S}$ with linear constant restrictions has a solution can be solved in* NEXPTIME *(and in time exponential in $|ST(\mathcal{S})|2^{|C|}$ by using resolution).*

(2) If $\mathcal{S}$ only contains the operation symbols $\wedge, 0, 1$, and, possibly, constants, then the problem can be decided in EXPTIME. *The same holds if $\mathcal{S}$ only contains the operation symbols $\vee, 0, 1$, and, possibly, constants.*

Proof: Similar to the proof of Theorem 4.4. □

4.3 The positive theory of D_{01}

By Proposition 2.5 and Theorem 2.6, the decidability of D_{01}-unification with linear constant restrictions implies the decidability of the positive theory of D_{01}. Moreover, the algorithms presented above for deciding D_{01}-unification with linear constant restrictions yield a decision algorithm for the positive theory of D_{01}.

With every positive sentence ϕ in disjunctive normal form

$$\phi = Q_1 x_1 \dots Q_m x_m (\bigvee_{i=1}^{q} (s_{i1} = t_{i1} \wedge \dots \wedge s_{in_i} = t_{in_i})),$$

we associate a set $\mathcal{S}_1, \dots, \mathcal{S}_q$ of q unification problems with linear constant restrictions, where $\mathcal{S}_i = \{s_{i1} = t_{i1}, \dots, s_{in_i} = t_{in_i}\}$ in which the existentially quantified variables are treated as variables, the universally quantified variables are treated as constants, and the linear order $x_1 < x_2 < \dots < x_m$ is induced by the quantifier prefix of ϕ.

THEOREM 4.10. *The D_{01}-validity of a positive sentence ϕ in disjunctive normal form can be solved in non-deterministically exponential time in the size of ϕ.*

Proof: Let $\phi = Q_1 x_1 \dots Q_m x_m (\bigvee_{i=1}^{q} (s_{i1} = t_{i1} \wedge \dots \wedge s_{in_i} = t_{in_i}))$ be a positive sentence in disjunctive normal form. By Proposition 2.5, ϕ is valid in all bounded distributive lattices if and only if one of the D_{01}-unification problems $\mathcal{S}_i = \{s_{i1} = t_{i1}, \dots, s_{in_i} = t_{in_i}\}$ (with linear constant restrictions lcr induced by the quantifier prefix of ϕ) has a solution with respect to D_{01}. It is therefore sufficient to check for all $1 \leq i \leq n$ whether $\mathcal{S}_i$ has a solution in D_{01}. By Theorem 4.9, the complexity of deciding whether $\mathcal{S}_i$ has a solution in D_{01} can be solved in at most non-deterministically exponential time in the size of $\mathcal{S}_i$. □

We are interested in automatically checking whether a positive sentence in disjunctive normal form is D_{01}-valid, or if a unification problem has a solution w.r.t. D_{01}. By Theorem 4.2 and Theorem 4.8 we know that the problems can be solved in doubly exponential time using resolution or a SAT checker. This is mainly due to the fact that exponentially many instances of otherwise simple clauses need to be analyzed. In [Sofronie-Stokkermans 2000] we introduced a schematic representation of these instances by using constrained clauses, and proved the soundness and completeness of the constrained resolution calculus obtained this way; however, as in this paper we focus on methods based on propositional satisfiability checking, the resolution calculus with constrained clauses is not presented in detail.

5. UNIFIABILITY *OVER* $F_{\mathsf{D}_{01}}(C)$

The notion of equation solving in an algebra can be generalized to the notion of equation solving *over* an algebra. This notion occurs in a natural way in the classical theory of fields, where we learn that every field can be extended to an algebraically closed field. In that context, if $\mathcal{S}$ is a system of equations it is not only interesting to know whether $\mathcal{S}$ has a solution in a field F, but also if it has a solution in the algebraic extension of F.

In problems related to applications (e.g. when testing unifiability of concept terms in description logics) it may not be sufficient to decide if a unification problem has a solution. It may also be important to know if it has a solution in some extension of the free algebra, or if additional operations (such as negation) are allowed. Consider the unification problem $\mathcal{S} = \{c \vee y = 1, c \wedge y = 0\}$. $\mathcal{S}$ has no solution in D_{01}, but it would have a solution in an extension of the free distributive lattice, in which every element has a complement (take $y = c'$, the complement of c).

Definition 5.1. Let $\mathcal{C}$ be a class of Σ-algebras. An algebra $\mathbf{A} \in \mathcal{C}$ is *algebraically closed* if every system of equations with parameters in $\mathbf{A}$ is satisfiable in some extension of $\mathbf{A}$ in $\mathcal{C}$ if and only if it is satisfiable in $\mathbf{A}$.

THEOREM 5.2. *A bounded distributive lattice is algebraically closed if and only if it is Boolean.*

Proof: We use a characterization of the algebraically closed distributive lattice given by [Schmid 1979], which states that a distributive lattice is algebraically closed if and only if it is relatively pseudocomplemented. The result now follows easily from the fact that a bounded distributive lattice is relatively pseudocomplemented if and only if it is Boolean.[7] □

We show that a unification problem is solvable over $F_{\mathsf{D}_{01}}(C)$ if and only if it is solvable in an algebraically closed (hence Boolean) extension of $F_{\mathsf{D}_{01}}(C)$. This will

[7]This result can also be proved directly, using a criterion that reduced checking whether D_{01} is algebraically closed to checking that a so-called "finite homomorphism property" holds for all duals of lattices in D_{01} (cf. e.g. [Clark and Davey 1998]).

allow to reduce testing whether a unification problem is solvable over $F_{\mathsf{D}_{01}}(C)$ to checking the truth (in the two-element lattice) of a quantified Boolean formula.

To prove this, we use some results on Priestley duality for bounded distributive lattices, which we summarize below. For further details on Priestley duality we refer to [Davey and Priestley 1992] and [Clark and Davey 1998].

Definition 5.3 *The Priestley dual of a bounded distributive lattice.* Let $\mathbf{L}$ be a bounded distributive lattice. Let $D(\mathbf{L}) := \mathsf{Hom}_{\mathsf{D}_{01}}(\mathbf{L}, \mathbf{2})$ be the set of all 0,1-lattice homomorphisms from $\mathbf{L}$ to the two-element bounded distributive lattice. The space $\mathbf{D}(\mathbf{L}) = (D(\mathbf{L}), \leq, \tau)$, where $\leq$ is the pointwise ordering on maps and τ is the topology generated by all sets of the form $X_a = \{h \in D(\mathbf{L}) \mid h(a)=1\}$ and their complements as a subbasis, is called the *Priestley dual* of $\mathbf{L}$.

If the lattice $\mathbf{L}$ is finite then the topology on $\mathbf{D}(\mathbf{L})$ is discrete, and the Priestley and Birkhoff duals coincide. If $\mathbf{L}$ is a Boolean lattice then the order on $\mathbf{D}(\mathbf{L})$ is discrete (no different elements are comparable).

THEOREM 5.4 [PRIESTLEY 1972]. *Let $\mathbf{L}$ be a bounded distributive lattice, let $\mathbf{D}(\mathbf{L})$ be its Priestley dual, and let $\mathbf{E}(\mathbf{D}(\mathbf{L})) = \mathsf{Hom}_P(\mathbf{D}(\mathbf{L}), \underline{\mathbf{2}})$ be the lattice of all continuous and order-preserving maps between the ordered topological space $\mathbf{D}(\mathbf{L})$, and the two-element partially-ordered set $\underline{\mathbf{2}}$ with the discrete topology. The map $\eta_{\mathbf{L}} : \mathbf{L} \to \mathbf{E}(\mathbf{D}(\mathbf{L}))$ defined for every $x \in L$, by $\eta_{\mathbf{L}}(x) : \mathbf{D}(\mathbf{L}) \to \underline{\mathbf{2}}$ (where, if $h \in \mathbf{D}(\mathbf{L}) = \mathsf{Hom}_{\mathsf{D}_{01}}(\mathbf{L}, \mathbf{2})$, $\eta_{\mathbf{L}}(x)(h) = h(x)$) is an isomorphism of bounded lattices.*

LEMMA 5.5. *Let C be a finite set, and let $F_{\mathsf{D}_{01}}(C)$ be the free bounded distributive lattice over C. Let $\mathbf{A}$ be an algebraically closed bounded distributive lattice, and let $i : F_{\mathsf{D}_{01}}(C) \hookrightarrow \mathbf{A}$ be an embedding. Then there exists an embedding $j : \mathcal{P}(D(F_{\mathsf{D}_{01}}(C))) \hookrightarrow \mathbf{A}$.*

Proof: Since $i : F_{\mathsf{D}_{01}}(C) \hookrightarrow \mathbf{A}$ is an embedding, by the Priestley duality theorem [Davey and Priestley 1992] $\mathbf{D}(i) : \mathbf{D}(\mathbf{A}) \to \mathbf{D}(F_{\mathsf{D}_{01}}(C))$ defined by $\mathbf{D}(i)(h) = h \circ i$ is a surjective continuous, order-preserving map. (Here $\mathbf{D}(\mathbf{L})$ is the Priestley dual of the lattice $\mathbf{L}$.) Since $\mathbf{A}$ is an algebraically closed bounded distributive lattice, it is Boolean, so the order on $\mathbf{D}(\mathbf{A})$ is discrete. Let $(X, \leq_d)$ be the space obtained from $\mathbf{D}(F_{\mathsf{D}_{01}}(C))$ by replacing the order $\leq$ by the discrete order $\leq_d$, where $x \leq_d y$ if and only if $x = y$. Let $j : \mathbf{D}(\mathbf{A}) \to (X, \leq_d)$ be defined by $j(y) = \mathbf{D}(i)(y)$ for every $y \in \mathbf{D}(\mathbf{A})$. It is easy to see that j is again a surjective continuous, order-preserving map, hence, by results on Priestley duality for distributive lattices, the map $\mathbf{E}(j) : \mathsf{Hom}_P((X, \leq_d), \underline{\mathbf{2}}) \to \mathsf{Hom}_P(\mathbf{D}(\mathbf{A}), \underline{2})$ defined by $\mathbf{E}(j)(h) = h \circ j$ is an embedding in D_{01}. The conclusion now follows from the fact that $\mathsf{Hom}_P((X, \leq_d), \underline{\mathbf{2}})$ is isomorphic to $\mathcal{P}(\mathbf{D}(F_{\mathsf{D}_{01}}(C)))$, and $\mathsf{Hom}_P(\mathbf{D}(\mathbf{A}), \underline{2})$ is isomorphic to $\mathbf{A}$. □

THEOREM 5.6. *Let $\mathcal{S} : \{s_1 = t_1, \ldots, s_k = t_k\}$ be a unification problem in the signature of D_{01}, with free constants in C, where $C = \{c_1, \ldots, c_n\}$ is a finite set. The following are equivalent:*

(1) $\mathcal{S}$ is solvable over *$F_{\mathsf{D}_{01}}(C)$;*

(2) there exists an algebraically closed extension $\mathbf{B}$ of $F_{\mathsf{D}_{01}}(C)$ such that $\mathcal{S}$ is solvable in $\mathbf{B}$;

(3) $\mathcal{S}$ is solvable in the Boolean lattice $(\mathcal{P}(X), \cup, \cap, \emptyset, X)$, where $X = \mathcal{P}(C)$;

(4) the formula $\forall c_1, \ldots, c_n \exists y_1 \ldots \exists y_m (s_1 = t_1 \wedge \cdots \wedge s_k = t_k)$ *is true in the two-element lattice* $\mathbf{2}$.

Proof: (1) $\Rightarrow$ (2) Assume that $\mathcal{S}$ is solvable *over* $F_{\mathsf{D}_{01}}(C)$. Then there exists an algebra $\mathbf{A} \in \mathsf{D}_{01}$ and an embedding $F_{\mathsf{D}_{01}}(C) \hookrightarrow \mathbf{A}$ such that $\mathcal{S}$ is solvable in $\mathbf{A}$. We know that every bounded distributive lattice can be embedded into a Boolean lattice. Hence, there exists a Boolean lattice $\mathbf{B}$ which extends $\mathbf{A}$. By Theorem 5.2, $\mathbf{B}$ is algebraically closed. Moreover, since $\mathcal{S}$ is solvable in $\mathbf{A}$, and $\mathbf{B}$ extends $\mathbf{A}$, it immediately follows that $\mathcal{S}$ is also solvable in $\mathbf{B}$.

(2) $\Rightarrow$ (3) Let $\mathbf{B}$ be an algebraically closed extension of $F_{\mathsf{D}_{01}}(C)$ in which $\mathcal{S}$ is solvable. By Lemma 5.5 we know that there exists an embedding $j : \mathcal{P}(D(F_{\mathsf{D}_{01}}(C))) \hookrightarrow \mathbf{B}$. Since $\mathcal{P}(D(F_{\mathsf{D}_{01}}(C)))$ is Boolean, it is also algebraically closed, hence $\mathcal{S}$ is solvable in $\mathcal{P}(D(F_{\mathsf{D}_{01}}(C)))$. The conclusion now follows easily from the fact that $\mathcal{P}(D(F_{\mathsf{D}_{01}}(C)))$ and $\mathcal{P}(X)$, where $X = \mathcal{P}(C)$, are isomorphic lattices.

(3) $\Rightarrow$ (1) follows from the fact that $i \circ \eta_C : F_{\mathsf{D}_{01}}(C) \overset{\eta_C}{\hookrightarrow} \mathcal{O}(X) \overset{i}{\subseteq} \mathcal{P}(X)$, where $X = \mathcal{P}(C)$, is an embedding.

In order to prove the equivalence of (3) and (4), note that $\tau : \mathcal{P}((\mathcal{P}(C), \subseteq)) \to \prod_{g:C \to \{0,1\}} \mathbf{2}$, defined for every $U \subseteq \mathcal{P}(C)$ by $\tau(U) = (\chi_U(g^{-1}(1)))_{g:C\to\{0,1\}}$, where χ_U is the characteristic function of U, is an isomorphism of bounded lattices. We will denote by $\pi_g : \prod_{g:C\to\{0,1\}} \mathbf{2} \to \mathbf{2}$ the projection on the component g.

(3) $\Rightarrow$ (4) If $\mathcal{S}$ is solvable in the Boolean lattice $\mathcal{P}(\mathcal{P}(C), \subseteq)$ then $\mathcal{S}$ is solvable in $\prod_{g:C\to\{0,1\}} \mathbf{2}$, i.e. there exists $h : Y \to \prod_{g:C\to\{0,1\}} \mathbf{2}$ such that $\overline{h}(s_i) = \overline{h}(t_i)$ for every $1 \leq i \leq k$, where $\overline{h}$ is the unique homomorphism that extends h, with the property that $\overline{h}(c) = (g(c))_{g:C\to\{0,1\}}$. Let $g : C \to \{0,1\}$, and let $h_g : Y \cup C \to \{0,1\}$ be defined by $h_g(c) := g(c)$ for every $c \in C$ and $h_g(y) := \pi_g \circ h$ for every $y \in Y$. It is easy to see that $\overline{h}_g = \pi_g \circ \overline{h}$. Therefore, $\overline{h}_g(s_i) = \overline{h}_g(t_i)$ for all $1 \leq i \leq k$. This shows that $\forall c_1, \ldots, c_n \exists y_1, \ldots, y_m (s_1 = t_1 \wedge \cdots \wedge s_k = t_k)$ is true in the two-element lattice.

(4) $\Rightarrow$ (3) Assume that the formula $\forall c_1, \ldots, c_n \exists y_1 \ldots \exists y_m (s_1 = t_1 \wedge \cdots \wedge s_k = t_k)$ is true in the two-element lattice $\mathbf{2}$. This means that for every $g : C \to \{0,1\}$ there exists $h_g : Y \to \{0,1\}$ such that $\overline{h}_g(s_i) = \overline{h}_g(t_i)$ for all $1 \leq i \leq k$, where $\overline{h}_g$ is the unique homomorphism that extends h_g, with the property that $\overline{h}_g(c) = g(c)$ for every $c \in C$. Let $h : Y \to \prod_{g:C\to\{0,1\}} \mathbf{2}$ be defined by $h(y) = (h_g(y))_{g:C\to\{0,1\}}$, and let $\overline{h} : \mathsf{Term}(Y \cup C) \to \prod_{g:C\to\{0,1\}} \mathbf{2}$ be the unique extension of h to a homomorphism, with the additional property that $\overline{h}(c) = (g(c))_{g:C\to\{0,1\}}$. It is easy to see that $\overline{h}(t) = (h_g(t))_{g:C\to\{0,1\}}$. Hence, for every $1 \leq i \leq k$, $\overline{h}(s_i) = (h_g(s_i))_{g:C\to\{0,1\}} = (h_g(t_i))_{g:C\to\{0,1\}} = \overline{h}(t_i)$. $\square$

Remark 5.7. From the equivalence of (1) and (3) in Theorem 5.6 it immediately follows that the unification problem $\mathcal{S} : \{s_1 = t_1, \ldots, s_k = t_k\}$ is solvable *over* $F_{\mathsf{D}_{01}}(C)$ if and only if the conjunction of the following formulae (in a signature which contains unary predicate symbols $\{P_e \mid e \in ST(\mathcal{S})\}$, a constant D for every subset D of C, and no other function symbols) is satisfiable:

$$
\begin{array}{lllll}
(\mathsf{Ren}) & (\wedge\mathsf{n}) & P_{e_1 \wedge e_2}(X) \rightarrow P_{e_i}(X) & \text{for all } X \subseteq C, i = 1,2 \\
 & (\wedge\mathsf{p}) & P_{e_1}(X) \wedge P_{e_2}(X) \rightarrow P_{e_1 \wedge e_2}(X) & \text{for all } X \subseteq C \\
 & (\vee\mathsf{n}) & P_{e_1 \vee e_2}(X) \rightarrow P_{e_1}(X) \vee P_{e_2}(X) & \text{for all } X \subseteq C \\
 & (\vee\mathsf{p}) & P_{e_i}(X) \rightarrow P_{e_1 \vee e_2}(X) & \text{for all } X \subseteq C, i = 1,2 \\
 & (1) & P_1(X) & \text{for all } X \subseteq C \\
 & (0) & \neg P_0(X) & \text{for all } X \subseteq C \\
 & (\mathsf{cp}) & P_c(X) & \text{for all } X \subseteq C \text{ with } c \in X \\
 & (\mathsf{cn}) & \neg P_c(X) & \text{for all } X \subseteq C \text{ with } c \notin X \\
(\mathsf{P}) & & P_{s_i}(X) \leftrightarrow P_{t_i}(X), & \text{for all } X \subseteq C, \text{ for all } 1 \leq i \leq k
\end{array}
$$

where each formula in (Ren) ∪ (P) is the conjunction of all formulae obtained by instantiating the variables X, resp. X_1, X_2 with subsets of C (which can be considered to be constants) satisfying the additional conditions: the indices $e_1 \vee e_2, e_1 \wedge e_2, 0, 1, c$ range over all elements in $ST(\mathcal{S})$; y ranges over all variables in $\{y_1, \dots, y_n\}$.

This conjunction is obtained from (Her)∪(Ren)∪(P) in Theorem 4.2, by omitting (Her). Due to the absence of the formulae (Her), which establish links between the truth value of $P_e(X)$ and $P_e(Y)$ if $X \subseteq Y$, (Ren) ∪ (P) can be regarded as the conjunction of $|\mathcal{P}(C)|$ independent satisfiability problems of size linear in $|C|$, one for each $X \subseteq C$. Checking the satisfiability of (Ren)∪(P) can be reduced to checking these independent problems. This is the same as checking the Boolean satisfiability of the formulae obtained from the QBF in (4) by substituting combinations of values in $\{0,1\}$ for the constants in C, and provides an alternative justification for the reduction of (1) to the validity of the quantified Boolean formula in (4).

We analyze the complexity of unification with free constants in D_{01} over the free algebra. For this, we use a reduction to the validity problem for *AE* quantified Boolean formulae (*AE QBF*), i.e. formulae of the form $(\forall x_1 \dots x_n)(\exists y_1 \dots y_m)E$, where E is a Boolean expression containing the propositional variables $x_1, \dots, x_n$, and $y_1, \dots, y_m$. The validity problem for AE QBF is Π_2^p-complete (cf. e.g. [Garey and Johnson 1979]).

THEOREM 5.8. *The unification problem with free constants in* D_{01} over *the free algebra is* Π_2^p*-complete.*

Proof: Let $\mathcal{S} = \{s_1 = t_1, \dots, s_k = t_k\}$ be an unification problem with free constants $C = \{c_1, \dots, c_n\}$ and variables $\{y_1, \dots, y_m\}$. By Theorem 5.6, we know that $\mathcal{S}$ is solvable *over* $F_{\mathsf{D}_{01}}(C)$ if and only if the formula

$$\forall c_1, \dots, c_n \exists y_1 \dots \exists y_m (s_1 = t_1 \wedge \dots \wedge s_k = t_k) \tag{2}$$

is true in the two-element lattice $\{0,1\}$. Therefore we can restrict to the study of validity of problems of the second type.

The problem is obviously in Π_2^p, since formula (2) is true in the two-element lattice $\{0,1\}$ if and only if the following AE QBF problem is valid:

$$\forall c_1, \dots, c_n \exists y_1 \dots \exists y_m (\bigwedge_{i=1}^{k} (s_i \leftrightarrow t_i) = 1).$$

Π_2^p-hardness can be proved as follows. Let E be a Boolean formula containing the propositional variables $x_1, \dots, x_n$, $y_1, \dots, y_m$. Let E' be the negation normal form of E. E' can be obtained from E in polynomial time. In E' the negation sign is applied only to the propositional variables $x_1, \dots, x_n, y_1, \dots, y_m$. It is not difficult to check that $(\forall x_1 \dots x_n)(\exists y_1 \dots y_m)E$ is satisfiable if and only if

$$(\forall x_1 \dots x_n)(\exists x'_1 \dots x'_n)(\exists y_1 \dots y_m)(\exists y'_1 \dots y'_m)$$
$$\bigwedge_{i=1}^n (x_i \wedge x'_i = 0 \quad \wedge \quad x_i \vee x'_i = 1) \wedge \bigwedge_{i=1}^m (y_i \wedge y'_i = 0 \quad \wedge \quad y_i \vee y'_i = 1) \wedge E'' = 1$$

is satisfiable in the two element lattice, where E'' is obtained from E' by replacing every negated occurrence of x_i by x'_i and every negated occurrence of y_i by y'_i. $\Box$

6. EXAMPLES AND COMPARISONS

We illustrate the methods described above by examples, and compare the methods described in this paper with other possible ways of deciding unifiability in D_{01}.

6.1 Translation to clause form and resolution or SAT checking

We consider the following unification problems with respect to D_{01}:

(1) $\mathcal{S}_1 = \{y \wedge c = 0, y \vee c = 1\}$, corresponding to $\phi_1 = \forall c \exists y (y \wedge c = 0 \text{ and } y \vee c = 1)$ (false in D_{01}, since not all distributive lattices are boolean);

(2) $\mathcal{S}_2 = \{c_3 \wedge y = c_1 \wedge c_2, c_3 \vee y = c_1 \vee c_2\}$, corresponding to the formula $\phi_2 = \forall c_1 \forall c_2 \forall c_3 \exists y (c_3 \wedge y = c_1 \wedge c_2 \text{ and } c_3 \vee y = c_1 \vee c_2)$
(false in D_{01}, as can be seen by taking $c_1 = c_2 = 0$ and $c_3 = 1$);

(3) $\mathcal{S}_3 = \{y \wedge c_1 = 0, (y \vee c_1) \wedge c_2 = c_2\}$, with the linear constant restriction $y < c_2$, corresponding to the formula $\phi_3 = \forall c_1 \exists y \forall c_2 (c_1 \wedge y = 0 \text{ and } (c_1 \vee y) \wedge c_2 = c_2)$
(false in D_{01}, since for $c_2 = 1$ it would reduce to ϕ_1);

(4) $\mathcal{S}_4 = \{y \wedge c_1 = c_1, (y \vee c_1) \wedge c_2 = c_2 \wedge y\}$, with the linear constant restriction $y < c_2$, corresponding to $\forall c_1 \exists y \forall c_2 (c_1 \wedge y = c_1 \text{ and } (c_1 \vee y) \wedge c_2 = c_2 \wedge y)$
(true in D_{01} as can be seen by taking $y = 1$);

(5) $\mathcal{S}_5 = \{y \wedge c_1 = 0, y \vee c_2 = c_2\}$, with the linear constant restriction $y < c_2$, corresponding to the formula $\forall c_1 \exists y \forall c_2 (c_1 \wedge y = 0 \text{ and } y \wedge c_2 = c_2)$
(true in D_{01} as can be seen by taking $y = 0$);

(6) $\mathcal{S}_6 = \{(c_1 \vee y) \wedge c_2 = y, (c_2 \vee y) \wedge c_3 = y\}$, corresponding to the formula $\forall c_1 \forall c_2 \forall c_3 \exists y ((c_1 \vee y) \wedge c_2 = y \text{ and } (c_2 \vee y) \wedge c_3 = y)$
(false in D_{01}, since it implies $c_2 \wedge (c_1 \vee c_3) = c_2 \wedge c_3$);

(7) $\mathcal{S}_7 = \{(c_1 \vee y) \wedge c_2 = y, (c_2 \vee y) \wedge c_3 = c_2 \wedge c_3\}$, corresponding to the formula $\forall c_1 \forall c_2 \forall c_3 \exists y ((c_1 \vee y) \wedge c_2 = y \text{ and } (c_2 \vee y) \wedge c_3 = c_2 \wedge c_3)$
(true in D_{01}, as can be seen by taking $y = c_2 \wedge c_3$);

(8) $\mathcal{S}_8 = \{(c_3 \vee c_4) \wedge y = c_0 \wedge c_1 \wedge c_2, (c_3 \vee c_4) \vee y = (c_0 \wedge c_1) \vee c_2\}$, corresponding to the formula $\forall x_0, x_1, x_2, x_3, x_4 \exists y ((x_3 \vee x_4) \wedge y = x_0 \wedge x_1 \wedge x_2,$ and $(x_3 \vee x_4) \vee y = (x_\wedge x_1) \vee x_2)$ (false in D_{01}, since for $c_3 = c_4$ and $c_0 = c_2$ we obtain $\mathcal{S}_2$);

(9) $\mathcal{S}_9 = \{y \wedge ((c_1 \vee c_3) \wedge c_4) = 0, y \vee c_2 = c_2\}$, with the linear constant restriction $y < c_2$, corresponding to the formula $\forall x_1, x_3, x_4 \exists y \forall x_2 (y \wedge ((x_1 \vee x_3) \wedge x_4) = 0$ and $y \vee x_2 = x_2)$ (true in D_{01});

(10) $\mathcal{S}_{10} = \{(c_1 \vee c_0) \vee y = c_0 \vee c_1, c_2 \vee (c_3 \wedge c_4) = z, (c_1 \vee c_0) \wedge y = y, (c_3 \wedge c_4) \wedge c_2 = u\}$, corresponding to the formula $\forall x_0, x_1, x_2, x_3, x_4 \exists y, z, u \ ((x_1 \vee x_0) \vee y = x_1 \vee x_0,$ and $x_2 \vee (x_3 \wedge x_4) = z,$ and $(x_1 \vee x_0) \wedge y = y,$ and $x_2 \wedge (x_3 \wedge x_4) = u)$ (true in D_{01}).

Unification in D_{01}. We used two different methods for checking whether the problems above are D_{01}-unifiable, both based on the structure-preserving translation to clause form described in Section 4. The run tests are described in Table I.

The *first approach* **(FO)** uses first-order logic as much as possible:

- We reduce the explicit description of the "$\subseteq$" relationship on $\mathcal{P}(C)$ to a minimum: we specify (by means of first-order clauses) the reflexivity and transitivity of "$\subseteq$", and describe explicitly only the immediate subset relationships ($X \subseteq \{c\} \cup X$ where $c \notin X$). Similarily, we specify only a minimal set of instances of sets X, Y with $X \not\subseteq Y$, chosen such that the other instances of $\not\subseteq$ follow as consequences using transitivity of $\subseteq$.
- All other clauses are generated in first-order form (the arguments of the predicates $P_i(X)$ occur as universally quantified variables).

Translation to clause form is done by FLOTTER, and SPASS [Weidenbach et al. 1996] is used to check the satisfiability of the resulting set of clauses.

The *second* approach **(SAT)** is purely propositional: we use distinct propositional variables to encode the literals $P_e(X)$, for every $e \in ST(\mathcal{S})$ and every $X \subseteq C$, and explicitly specify all instances of $(\mathsf{Her}) \cup (\mathsf{Ren}) \cup (\mathsf{P})$ and, if necessary, (Lcr). The input does not need to be in clause form. Then FLOTTER [Weidenbach et al. 1996] is used to generate an optimized translation to clause form. The satisfiability of the resulting set of clauses is checked with SPASS and zChaff [Moskewicz et al. 2001]. In order to be able to use zChaff, we translated the set of clauses into DIMACS format after making an intermediate translation from .dfg into TPTP format.

	FO			**SAT**						
	SPASS			SPASS			zChaff			
Unif. probl.	# cl in	mem. (KB)	time (s)	# cl in	mem. (KB)	time (s)	# var	# cl. in	time (s)	status
$\mathcal{S}_1$	19	571	0.00	27	533	0.00	12	27	0.00	unsatisfiable
$\mathcal{S}_2$	36	596	0.01	164	624	0.01	64	164	0.00	unsatisfiable
$\mathcal{S}_3$	27	583	0.01	70	566	0.01	28	70	0.01	unsatisfiable
$\mathcal{S}_4$	29	590	0.01	66	564	0.01	28	66	0.00	satisfiable
$\mathcal{S}_5$	24	581	0.01	58	558	0.01	24	58	0.00	satisfiable
$\mathcal{S}_6$	36	596	0.01	164	625	0.01	64	164	0.00	unsatisfiable
$\mathcal{S}_7$	39	634	0.02	188	641	0.02	72	188	0.01	satisfiable
$\mathcal{S}_8$	108	1107	0.11	939	1108	0.13	384	939	0.00	unsatisfiable
$\mathcal{S}_9$	66	697	0.03	376	755	0.04	160	376	0.00	satisfiable
$\mathcal{S}_{10}$	114	1258	0.17	1217	1277	0.23	448	1217	0.00	satisfiable

where "# cl.in" is the number of clauses generated by Flotter from the input formulae, and "var" is the number of propositional variables in the input.

Table I. Unification in D_{01}: Some run examples

Unification over $F_{\mathsf{D}_01}(C)$. We now illustrate the methods sketched in Section 5 for checking unifiability *over* $F_{\mathsf{D}_01}(C)$ for the unification problems without linear constant restrictions, namely $\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_6, \mathcal{S}_7, \mathcal{S}_8$ and $\mathcal{S}_{10}$.

The *first method* (**SAT**) is based on Remark 5.7 in Section 5. We check the satisfiability of the (propositional) clauses obtained from the structure-preserving translation to clause form described in Section 4, from which the clauses (Her) expressing monotonicity have been removed. The satisfiability is checked both by resolution, using SPASS, and with the SAT checker zChaff. This method is not space efficient: as mentioned in Remark 5.7, the set of clauses obtained by the translation mentioned above can be written as a union of $2^{|C|}$ independent set of clauses over disjoint sets of propositional variables. The satisfiability of these independent sets of clauses can be checked also separately: in Table II, the run examples of this refinement are $\mathcal{S}_i(\text{dec}, k)$, where $k = 2^{|C|}$ is the number of independent problems, each with the same number of clauses. (In this last case, the memory and time entries indicate the average memory and time for the individual subproblems.)

The *second method* (**QBF**) is based on Theorem 5.6, where the problem of deciding the solvability of a unification problem over $F_{\mathsf{D}_{01}}(C)$ is reduced to the problem of testing the truth of a QBF. A naive method for translation to clause form was used (a structure-preserving translation to clause form would have increased the number of existentially quantified variables and would have yielded clauses similar to those generated with the first method, after decomposition). We made experiments with QuBE [Giunchiglia et al. 2001] and SEMPROP [Letz 2002].

The run tests (for SPASS, zChaff and QuBE) are described in Table II.

	SAT						**QBE**					
	SPASS			zChaff			QuBE (∀∃)					
Unif. probl.	# cl. in	mem. (KB)	time (s)	# var in	# cl. (KB)	time (s)	# var ∀	# var ∃	# cl. in	mem. (KB)	time (s)	status
$\mathcal{S}_1$	26	416	0.00	12	26	0.00	1	1	2	1.80	0.00	true
$\mathcal{S}_1$ (dec, 2)	13	383	0.00	6	13	0.00						
$\mathcal{S}_2$	152	457	0.01	64	152	0.00	3	1	8	2.74	0.00	false
$\mathcal{S}_2$ (dec, 8)	19	385	0.00	8	19	0.00						
$\mathcal{S}_6$	152	457	0.01	64	152	0.00	3	1	8	2.44	0.00	false
$\mathcal{S}_6$ (dec, 8)	19	385	0.00	8	19	0.00						
$\mathcal{S}_7$	176	466	0.01	72	176	0.00	3	1	10	2.26	0.00	true
$\mathcal{S}_7$ (dec, 8)	22	386	0.00	9	22	0.00						
$\mathcal{S}_8$	864	691	0.08	384	864	0.00	5	1	12	3.57	0.00	false
$\mathcal{S}_8$ (dec, 32)	27	387	0.00	12	27	0.00						
$\mathcal{S}_{10}$	992	739	0.13	448	992	0.00	5	3	16	3.60	0.00	true
$\mathcal{S}_{10}$ (dec, 32)	31	388	0.00	14	31	0.00						

where "# cl.in" is the number of clauses generated by Flotter from the input formulae, and "var" is the number of propositional variables in the input.

Table II. Unification over $F_{\mathsf{D}_{01}}(C)$: Some run examples

6.2 Comparisons

We now present several other methods for testing whether a unification problem has a solution. They are either less efficient or less general than the method we presented in this paper.

Equational reasoning. If $\mathcal{S} = \{s_1 = t_1, \ldots, s_n = t_n\}$ is a unification problem with linear constant restrictions lcr, one possibility of proving that $\mathcal{S}$ has a solution would be to prove that the formula $Q_1 x_1 \ldots Q_n x_n (\bigwedge_{i=1}^n s_i = t_i)$ is a consequence of the axioms for bounded distributive lattices with operators, where $Q_1 x_1 \ldots Q_n x_n$ is the quantifier alternation which corresponds to lcr. We used SPASS to check whether the formulae associated with the unification problems $\mathcal{S}_1$, $\mathcal{S}_2$, and $\mathcal{S}_3$ are consequences of the lattice axioms, but execution did not terminate, even for a very simple example such as $\mathcal{S}_1$. This shows that the translation to first-order logic without equality which we describe in this paper reduces problems which are quite difficult, or even unfeasible, for existing theorem provers to very easy problems.

A straightforward method. Let $\mathcal{S} : \{s_1 = t_1, \ldots, s_k = t_k\}$ be a D_{01}-unification problem with free constants in a finite set C and variables in the finite set Y. We can check whether $\mathcal{S}$ has a solution by checking if there is an instantiation of the variables in $\mathcal{S}$ with elements in $F_{\mathsf{D}_{01}}(C)$ that satisfies $\mathcal{S}$. There exist at most $(2^{2^{|C|}})^{|Y|}$ such instantiations. For each instantiation $h : Y \to F_{\mathsf{D}_{01}}(C)$, one has to check if $h(s_i) \equiv_{D_{01}} h(t_i)$, $1 \leq i \leq k$. There exists an algorithm for disproving the equivalence of two terms which is nondeterministically polynomial in the length of the terms [Hunt et al. 1987]. The elements in $F_{\mathsf{D}_{01}}(C)$ can be written as disjunctions of conjunctions of elements in C; the length of such a term is at most $|C| \cdot 2^{|C|}$. Hence, the length of $h(s_i)$ and $h(t_i)$, $1 \leq i \leq k$, can at most be $|Y| \cdot |C| \cdot 2^{|C|} + \mathsf{max}(\mathcal{S})$, where $\mathsf{max}(\mathcal{S})$ is the maximal length of a term occurring in $\mathcal{S}$. This shows that the method above would require to solve $(2^{2^{|C|}})^{|Y|}$ problems whose complexity is, in the worst case, nondeterministically polynomial in $|Y| \cdot |C| \cdot 2^{|C|} + \mathsf{max}(\mathcal{S})$.

For instance, in order to check $\mathcal{S}_1$ we would need to check if there is an instantiation of y with elements in $F_{\mathsf{D}_{01}}(\{c\})$ (a 3-element algebra, with elements $0, c$, and 1) that satisfies $\mathcal{S}_1$. This amounts to deciding the following conjunctions of word problems: ($0 \wedge c = 0$ and $0 \vee c = 1$), ($c \wedge c = 0$ and $c \vee c = 1$) and ($1 \wedge c = 0$ and $1 \vee c = 1$).

However, for example $\mathcal{S}_2$ this approach is very inefficient. The structure of the free lattice in 3 generators, $F_D(\{c_1, c_2, c_3\})$, is known; it has been shown that it has 18 elements, so the free lattice in D_{01} has 20 elements. Thus, if the description of $F_D(\{c_1, c_2, c_3\})$ is assumed to be known, then the algorithm described above would require solving 20 word problems in the variables c_1, c_2, c_3. If the description is not known, it would be necessary to solve 2^{2^3} word problems in the variables c_1, c_2, c_3, one for each possible instantiation of y by a conjunction of disjunctions of elements in $\{c_1, c_2, c_3\}$.

In fact, the precise structure of $F_{D_{01}}(C)$ is known only for small values of $|C|$. Therefore, in general, one would really need to verify all possible conjunctions of disjunctions of elements in C (the number could possibly be reduced slightly by exploiting the absorption property in lattices).

A special case. In [Gerhard and Petrich 1994] Gerhard and Petrich present the following criterion for unifiability for one single equation, i.e. for the unification problem $\mathcal{S} : \{s = t\}$.

(1) Let s' and t' be the disjunctive normal forms of s resp. t.
(2) If neither s' or t' has a constant term[8] then s and t are unifiable.
(3) If s' or t' has constant terms, let $h\colon Y \rightarrow T_{D_{01}}(C)$ be defined by $h(x) = D$ for every $x \in Y$, where D is the disjunction of all constant terms in s' and t'.
(4) If $\overline{h}(s) \equiv_{D_{01}} \overline{h}(t)$ then s and t are unifiable, otherwise they are not unifiable.

The disjunction D in Step 3 can be determined in polynomial time with respect to $\mathsf{length}(s') + \mathsf{length}(t')$. The same holds for the process of replacing every variable in s and t by D. Both the length of D and the length of the result of replacing all variables in s, t by D is polynomial in $\mathsf{length}(s') + \mathsf{length}(t')$; but may be exponential in $\mathsf{length}(s) + \mathsf{length}(t)$. Hence, the complexity of the criterion above is given by the complexity of Step 1 (computing the disjunctive normal forms of s and t) and Step 4 (solving a word problem). The last problem is co-NP complete [Hunt et al. 1987]; there exists an algorithm for disproving the equivalence of $\overline{h}(s)$ and $\overline{h}(t)$ which is nondeterministically polynomial in $\mathsf{length}(\overline{h}(s)) + \mathsf{length}(\overline{h}(t))$. Together with the fact that the lengths of $\overline{h}(s)$ and $\overline{h}(t)$ may be exponential in $\mathsf{length}(s) + \mathsf{length}(t)$ (in fact, in the worst case it can be of order $(\mathsf{length}(s) + \mathsf{length}(t)) \cdot 2^{|C|}$), this shows that the algorithm described above is non-deterministically polynomial in $(\mathsf{length}(s) + \mathsf{length}(t)) \cdot 2^{|C|}$.

Note that the method of Gerhard and Petrich above uses the fact that the unification problem consists of *only one equation*. It seems unlikely to us that the idea can be extended to deal with problems consisting of several equations, or to dealing with linear constant restrictions.

7. CONCLUSION

We presented a method for deciding unifiability with respect to the equational theory of bounded distributive lattices with operators. The method uses the Birkhoff representation for finite bounded distributive lattices, in particular the description of the dual $\mathbf{D}(F_{D_{01}}(C))$ of the free lattice in D_{01} over C as $(\mathcal{P}(C), \subseteq)$.

This helped us to reduce the problem of checking whether a D_{01}-unification problem $\mathcal{S}$ with constants C (and linear constraint restrictions) has a solution, to the problem of checking the satisfiability of a set $\Phi_{\mathcal{S}}$ of clauses. $\Phi_{\mathcal{S}}$ can be represented both as a finite set of ground clauses, and as a set of constrained clauses. The satisfiability of $\Phi_{\mathcal{S}}$ can be tested with any theorem prover or SAT solver.

The algorithm we propose is in NEXPTIME. However, SAT checkers are nowadays very efficient, even for problems with many propositional variables. Moreover, we showed that sometimes syntactic information about the terms in $\mathcal{S}$ is reflected by the form of clauses, which allows us to establish better upper bounds for particular classes of problems. In [Sofronie-Stokkermans 2000] we introduced a schematic representation of the ground instances of the problems obtained this way by using

[8] If $s = \bigvee_i \bigwedge_{j \in I_i} s_j$ is in disjunctive normal form, then a constant term of s is any of the conjunctions $\bigwedge_{j \in I_i} s_j$ in s not containing any variable.

constrained clauses and formulated a sound and complete resolution calculus for constrained clauses. This method is still exponential in $|ST(\mathcal{S})|2^C$, and is not discussed in this paper.

We also analyzed the problem of deciding unifiability *over* the free algebra (i.e. in an algebraic extension) for the equational theory of D_{01}, and proved that this reduces to Boolean unifiability. We would like to compare our method with more general unification algorithms, e.g. based on rewriting, which decide unifiability *over* the free algebra (e.g. the one in [Bockmayr 1992]); we expect that the need of taking into account the lattice axioms might make the rewriting-based approach less efficient due to the necessity of dealing with axioms such as associativity and commutativity.

Acknowledgments. I thank Harald Ganzinger for drawing my attention to the results on unification with linear constant restrictions [Baader and Snyder 2001] and on Boolean unification [Baader and Narendran 2001]. I thank Jean-Marc Talbot for the discussion we had on possible applications of unification to set constraints. I thank the referees for their helpful suggestions.

REFERENCES

Baader, F. and Narendran, P. 2001. Unification of concepts terms in description logics. *Journal of Symbolic Computation 31,* 3, 277–305.

Baader, F. and Schulz, K. 1996. Unification in the union of disjoint equational theories: Combining decision procedures. *Journal of Symbolic Computation 21*, 211–243.

Baader, F. and Schulz, K. 1998. Combination of constraint solvers for free and quasi-free structures. *Theoretical Computer Science 192*, 107–161.

Baader, F. and Snyder, W. 2001. Unification theory. In *Handbook of Automated Reasoning*, A. Robinson and A. Voronkov, Eds. Vol. I. Elsevier Science, Chapter 8, 445–532.

Birkhoff, G. 1933. On the combination of subalgebras. *Proceedings of the Cambridge Philosophical Society 29*, 441–464.

Bockmayr, A. 1992. Model-theoretic aspects of unification. In *Proceedings of Word Equations and Related Topics IWWERT'90, Tübingen, October 1990*, K. Schulz, Ed. LNCS 572. Springer Verlag, 181–196.

Bürckert, H. 1991. *A resolution principle for a logic with restricted quantifiers.* LNAI 568. Springer Verlag.

Burris, S. 1995. Polynomial time uniform word problems. *Mathematical Logic Quarterly 41*, 173–182.

Burris, S. and McKenzie, R. 1981. *Decidability and Boolean representations.* Memoirs of the American Mathematical Society series, vol. 32/246. American Mathematical Society, Providence, RI.

Clark, D. M. and Davey, B. A. 1998. *Natural dualities for the working algebraist.* Cambridge studies in advanced mathematics, vol. 57. Cambridge University Press.

Davey, B. and Priestley, H. 1992. *Introduction to lattices and order.* Cambridge Univ. Press, Cambridge.

Dowling, W. and Gallier, J. 1984. Linear-time algorithms for testing the satisfiability of propositional Horn formulae. *Journal of Logic Programming 1,* 3, 267–284.

Garey, M. and Johnson, D. 1979. *Computers and intractability : a guide to the theory of NP-completeness.* Mathematical sciences series. Freeman.

Gerhard, J. and Petrich, M. 1994. Unification in free distributive lattices. *Theoretical Computer Science 126,* 2, 237–257.

Ghilardi, S. 1997. Unification through projectivity. *Journal of Logic and Computation 7,* 6, 733–752.

GIUNCHIGLIA, E., NARIZZANO, M., AND TACCHELLA, A. 2001. QuBE a system for deciding quantified Boolean formulas satisfiability. In *Automated Reasoning, First International Joint Conference, IJCAR 2001, Proceedings*, R. Goré, A. Leitsch, and T. Nipkow, Eds. LNAI 2083. Springer Verlag, 364–369.

HUNT, H., ROSENKRANTZ, D., AND BLONIARZ, P. 1987. On the computational complexity of algebra of lattices. *SIAM Journal of Computation 16,* 1, 129–148.

HUNT, H. AND STEARNS, R. 1990. The complexity of very simple Boolean formulas with applications. *SIAM Journal of Computation 19,* 1, 44–70.

KIRCHNER, C. AND KIRCHNER, H. 1989. Constrained equational reasoning. In *Proceedings of the ACM-SIGSAM 1989 International Symposium on Symbolic and Algebraic Computation, Portland (Oregon).* ACM Press, 382–389. Report CRIN 89-R-220.

LETZ, R. 2002. Lemma and model caching in decision procedures for quantified Boolean formulas. In *Proceedings of the International Conference on Automated Reasoning with Analytic Tableaux and Related Methods (TABLEAUX 2002)*, C. Fermüller and U. Egly, Eds. LNAI 2381. Springer Verlag, 160–175.

MCKINSEY, J. 1943. The decision problem for some classes of sentences without quantifiers. *The Journal of Symbolic Logic 8,* 3, 61–76.

MOSKEWICZ, M., MADIGAN, C., ZHAO, Y., ZHANG, L., AND MALIK, S. 2001. Chaff: Engineering an efficient SAT solver. In *Proceedings of the 39th Design Automation Conference (DAC 2001).* ACM, 530–535.

NARENDRAN, P. 1996. Unification modulo ACI + 1 + 0. *Fundamementa Informaticae 25,* 1, 49–57.

NIEUWENHUIS, R. AND RUBIO, A. 1992. Theorem proving with ordering constrained clauses. In *Proceedings of CADE-11*, D. Kapur, Ed. LNAI 607. Springer Verlag, 477–491.

PRIESTLEY, H. 1972. Ordered topological spaces and the representation of distributive lattices. *Proc. London Math. Soc. 3*, 507–530.

SCHMID, J. 1979. Algebraically and existentially closed distributive lattices. *Zeitschrift für mathematische Logik und Grundlagen der Informatik 25*, 525–530.

SCHMIDT-SCHAUSS, M. 1998. A decision algorithm for distributive unification. *Theoretical Computer Science 208,* 1–2, 111–148.

SOFRONIE-STOKKERMANS, V. 1999. On the universal theory of varieties of distributive lattices with operators: Some decidability and complexity results. In *Proceedings of the 16th International Conference on Automated Deduction (CADE-16)*, H. Ganzinger, Ed. LNAI 1632. Springer Verlag, 157–171.

SOFRONIE-STOKKERMANS, V. 2000. On unification for bounded distributive lattices. In *Proceedings of the 17th International Conference on Automated Deduction (CADE-17)*, D. McAllester, Ed. LNAI 1831. Springer Verlag, 465–481.

SOFRONIE-STOKKERMANS, V. 2003. Resolution-based decision procedures for the universal theory of some classes of distributive lattices with operators. *Journal of Symbolic Computation 36,* 6, 891–924.

SOFRONIE-STOKKERMANS, V. 2004. Resolution-based decision procedures for the positive theory of some finitely generated varieties of algebras. In *Proceedings of the 34th International Symposium on Multiple-Valued Logic (ISMVL-2004).* IEEE, IEEE Computer Society, 32–37.

WEIDENBACH, C., GAEDE, B., AND ROCK, G. 1996. SPASS & FLOTTER Version 0.42. In *Proceedings of CADE-13*, M. McRobie and J. Slaney, Eds. LNCS 1104. Springer Verlag, 141–145.

Received March 2003; revised December 2004 and April 2005; accepted May 2005