# A Solution of the Word Problem for Free Double Boolean Algebras

Björn Vormbrock

Technische Universität Darmstadt, Fachbereich Mathematik
Schloßgartenstr. 7, D-64289 Darmstadt,
vormbrock@mathematik.tu-darmstadt.de

**Abstract.** Double Boolean algebras were introduced in [Wi00a] as a variety fundamental for Boolean Concept Logic, an extension of Formal Concept Analysis allowing negations of formal concepts. In this paper, the free double Boolean algebra generated by the constants is described. Moreover, we show that every free double Boolean algebra with at least one generator is infinite. A measure of the complexity of terms specific for double Boolean algebras is introduced. This, together with a modification of the algorithm for protoconcept exploration (cf. [Vo04]) yields double Boolean algebras containing a counterexample to every term identity up to a given complexity if the identity does not hold in general. These algebras can be constructed automatically, thus the word problem for free double Boolean algebras is solved.

## 1 Introduction

Double Boolean algebras form the variety generated by protoconcept algebras. Protoconcept algebras were defined in order to introduce negations of concepts in Formal Concept Analysis and to develop a Boolean Concept Logic in the framework of Contextual Logic (see [Wi00b] for an introduction to Contextual Logic, [Wi00a] for an introduction to Boolean Concept Logic). For the development of Boolean Concept Logic, a solution to the word problem for the free double Boolean algebras is essential. In[HLSW00] Herrmann et al. show that double Boolean algebras have the finite embedding property. Since double Boolean algebras are finitely axiomatized it follows that their universal theory is decidable. Moreover, as every Boolean algebra may be regarded as a double Boolean algebra, this decision problem is $\mathcal{NP}$-complete.

In this paper, a measure for the compelxity of terms appropiate for double Boolean algebras is developed and an upper bound for the size of a minimal counterexample to an invalid term identity $s \sim t$ depending of the complexity of $s$ and $t$ is derived. Moreover, a class of protoconcept algebras containing all counterexamples for invalid term identities up to a given level of complexity is described. An algorithm for the automated construction of their underlying contexts is given. This provides a semantic solution of the word problem for free double Boolean algebras where the algorithm depends only of the complexity of the terms.

S.O. Kuznetsov and S. Schmidt (Eds.): ICFCA 2007, LNAI 4390, pp. 240–270, 2007.

In Section 1, the basic definitions for double Boolean algebras and protoconcept algebras are introduced. Moreover, the free double Boolean algebra generated by the constants is described. In Section 2, an algorithm for the stepwise exploration of double Boolean algebras is developed. The application of this algorithm to free double Boolean algebras in Section 3 yields a class of counterexamples to invalid term identities depending on the complexity of the terms and the number of variables. An upper bound for the size of a minimal counterexample is derived in Corollary 4. Section 4 is dedicated to the automated generation of the contexts underlying the described class of counterexamples.

### 1.1 Double Boolean Algebras and Protoconcept Algebras

**Double Boolean Algebras**

**Definition 1.** *A* double Boolean algebra *is an algebra* $\underline{D} := (D, \sqcap, \sqcup, \neg, \lrcorner, \bot, \top)$ *of type (2,2,1,1,0,0), satisfying the equations*

| | |
|---|---|
| *1a)* $(x \sqcap x) \sqcap y = x \sqcap y$ | *1b)* $(x \sqcup x) \sqcup y = x \sqcup y$ |
| *2a)* $x \sqcap y = y \sqcap x$ | *2b)* $x \sqcup y = y \sqcup x$ |
| *3a)* $x \sqcap (y \sqcap z) = (x \sqcap y) \sqcap z$ | *3b)* $x \sqcup (y \sqcup z) = (x \sqcup y) \sqcup z$ |
| *4a)* $x \sqcap (x \sqcup y) = x \sqcap x$ | *4b)* $x \sqcup (x \sqcap y) = x \sqcup x$ |
| *5a)* $x \sqcap (x \vee\!\!\!\!\sqcup y) = x \sqcap x$ | *5b)* $x \sqcup (x \overline{\sqcap} y) = x \sqcup x$ |
| *6a)* $x \sqcap (y \vee\!\!\!\!\sqcup z) = (x \sqcap y) \vee\!\!\!\!\sqcup (x \sqcap z)$ | *6b)* $x \sqcup (y \overline{\sqcap} z) = (x \sqcup y) \overline{\sqcap} (x \sqcup z)$ |
| *7a)* $\neg\neg(x \sqcap y) = x \sqcap y$ | *7b)* $\lrcorner\lrcorner(x \sqcup y) = x \sqcup y$ |
| *8a)* $\neg(x \sqcap x) = \neg x$ | *8b)* $\lrcorner(x \sqcup x) = \lrcorner x$ |
| *9a)* $x \sqcap \neg x = \bot$ | *9b)* $x \sqcup \lrcorner x = \top$ |
| *10a)* $\neg\bot = \top \sqcap \top$ | *10b)* $\lrcorner\top = \bot \sqcup \bot$ |
| *11a)* $\neg\top = \bot$ | *11b)* $\lrcorner\bot = \top$ |
| *12)* $(x \sqcap x) \sqcup (x \sqcap x) = (x \sqcup x) \sqcap (x \sqcup x)$ | |

*with the operations* $\vee\!\!\!\!\sqcup, \overline{\sqcap}, \overline{\top}, \underline{\bot}$ *defined by*

$$x \vee\!\!\!\!\sqcup y := \neg(\neg x \sqcap \neg y)$$
$$x \overline{\sqcap} y := \lrcorner(\lrcorner x \sqcup \lrcorner y)$$
$$\overline{\top} := \neg\bot$$
$$\underline{\bot} := \lrcorner\top$$

*A* pure double Boolean algebra *is a double Boolean algebra that satisfies the additional condition*

$$\textit{13)}\ x = x \sqcap x \text{ or } x = x \sqcup x.$$

To shorten notation we write $x_\sqcap$ for $x \sqcap x$ and $x_\sqcup$ for $x \sqcup x$, and define $D_\sqcap := \{x_\sqcap \mid x \in D\}$, $D_\sqcup := \{x_\sqcup \mid x \in D\}$ and $D_p := D_\sqcap \cup D_\sqcup$. The restriction of $\underline{D}$ to $D_p$ is a pure subalgebra of $\underline{D}$.

On double Boolean algebras we define a quasi-order $\sqsubseteq$ by:

$$x \sqsubseteq y :\Leftrightarrow x \sqcap y = x_\sqcap \text{ and } x \sqcup y = y_\sqcup.$$