# The Equational Theory of a Nontrivial Discriminator Variety is co-NP-hard

Stanley Burris

Abstract. Discriminator varieties play a central role in the classification of decidable varieties; and they arise naturally in the study of algebraic logics. There are also important connections with the reduction of theorem proving to equational logic. In this paper we show, for any nontrivial discriminator variety, that the problem of determining if an equation holds in the variety is co-**NP**-hard.

## 1. Background

A *variety* is a class of algebras closed under homomorphic images, subalgebras, and products; equivalently, it is a class of algebras defined by a set of equations. A variety $\mathcal{V}$ is *generated* by a class $\mathcal{K}$ of algebras, written $\mathcal{V} = V(\mathcal{K})$, if $\mathcal{V}$ is the smallest variety satisfying $\mathcal{K} \subseteq \mathcal{V}$. We say that a variety $\mathcal{V}$ is *finitely generated* if it can be generated by some finite set of finite algebras. For example the variety of distributive lattices is finitely generated – it is generated by any nontrivial class of distributive lattices. (A class of algebras is *trivial* if the algebras in it have only one element in them.)

We are interested in the *equational theory* of a class $\mathcal{K}$ of algebras, written $\mathsf{Th}_{eq}(\mathcal{K})$, which is just the set of equations true of every algebra in $\mathcal{K}$. The problem of determining if $s \approx t \in \mathsf{Th}_{eq}(\mathcal{K})$ is also called the *equivalence problem* for $\mathcal{K}$. Since

$$\mathsf{Th}_{eq}(\mathcal{K}) = \mathsf{Th}_{eq}(V(\mathcal{K}))$$

holds, we simply say we are interested in the equational theories of varieties.

**Proposition 1.1.** *If $\mathcal{V}$ is a finitely generated variety with a finite language then $\mathsf{Th}_{eq}(\mathcal{K})$ is co-***NP***.*

*Proof.* Let $\mathcal{K}$ be a finite set of finite algebras generating $\mathcal{V}$. Then $\mathcal{V}$ does not satisfy the equation $s \approx t$ iff $s \not\approx t$ is satisfiable in some member of $\mathcal{K}$. The latter is clearly a problem in **NP**. $\square$

For background on discriminator varieties the reader is referred to Burris [4] (1992) for a survey; and to Burris & Sankappanavar (1981), Chap. IV, for technical details.

*Date*: July 31, 2002.
This research was supported by a grant from NSERC.

A term $t(x, y, z)$ is a *discriminator term* for an algebra $\mathbf{A}$ if the corollaryresponding function $t^{\mathbf{A}}$ satisfies, for $a, b \in A$,

$$t^{\mathbf{A}}(a, b, c) = \begin{cases} a & \text{if} \quad a \neq b \\ c & \text{if} \quad a = b. \end{cases}$$

A variety $\mathcal{V}$ is a *discriminator variety* if there is a term $t(x, y, z)$ which is a discriminator term for all the subdirectly irreducibles in $\mathcal{V}$. The following are favorite examples of discriminator varieties.:

| | |
|---|---|
| $\mathcal{BA}$ | Boolean algebras |
| $\mathcal{BR}$ | Boolean rings |
| $\mathcal{RCDL}$ | relatively complemented distributive lattices |
| $\mathcal{P}_n$ | $n$-valued Post algebras |
| $x^n \approx x$-rings | (for $n > 1$) |
| $V(\mathbf{F}_1, \cdots, \mathbf{F}_n)$ | a variety generated by finite fields |
| $\mathcal{CA}_n$ | cylindric algebras of dimension $n$ |
| $\mathcal{RA}$ | relation algebras. |

All but the last two of the above examples are finitely generated discriminator varieties, and hence their equational theories are co-**NP**. Looking at the above list, the complexity of the equational theories has been previously determined (and published in the literature) for $\mathcal{BA}$, $\mathcal{BR}$, and $V(\mathbf{F}_1, \cdots, \mathbf{F}_n)$, where each $\mathbf{F}_i$ is a finite field. The co-**NP**-completeness of $\mathcal{BA}$ is essentially Cook's pioneering work on SAT. The variety $\mathcal{BR}$ and, more generally, $V(\mathbf{F})$ were proved co-**NP**-complete in Bloniarz, Hunt, & Rosenkrantz [2] (1984) (along with the variety generated by the ring $\mathbf{Z}_k$ of integers modulo $k$, for $k > 1$). The results of Hunt & Stearns [7] (1990) on nonnilpotent finite commutative rings covers the case of a variety generated by finitely many finite fields $\mathbf{F}_i$ (just consider the ring $\Pi_{i=1}^n \mathbf{F}_i$).

These, along with the "if-then-else" discriminator varieties studied in Bloom & Tindell [1] (1983) and Mekler & Nelson [9] (1987), are the basic examples studied in the literature.

Our technique for showing co-**NP**-hardness is to efficiently reduce distributive lattice equations to equations in the target discriminator variety. For this purpose we introduce some notation.

**1.1. The $(\bigvee, \bigwedge)$ - classes of lattice terms.** Because of the associative law it is common practice to drop numerous parentheses when writing out a lattice term $t$, for example,

$$x_1 \vee (((x_2 \wedge x_3) \wedge x_4) \vee x_5)$$

would be simply written as

$$x_1 \vee (x_2 \wedge x_3 \wedge x_4) \vee x_5.$$

Such expressions will be called *generalized* lattice terms. The translation of a lattice term into its generalized form will be denoted by $t^\star$. This translation leads to a classification of the $t$ [or $t^\star$]; in our example above we have a $\bigvee\bigwedge$ [generalized] term. Likewise one has $\bigvee\bigwedge\bigvee$ [generalized] terms, etc. The set of all lattice terms having the generalized form $\bigvee\bigwedge\bigvee$ is called the $\bigvee\bigwedge\bigvee$ - class; this is an example of a $(\bigvee,\bigwedge)$ - class.

One can refine this classification by adding $\bigvee_n$, $\bigvee_{\leq n}$, etc., to indicate exactly $n$, or $\leq n$, arguments in the generalized operations. The class $\bigwedge\bigvee_3$ is important because of the use made of 3-SAT.

For computer implementations we could express the generalized form $t^\star$ of a term $t$ by a list; in our example above we could use

$$\Big(\bigvee x_1\ (\bigwedge x_2\ x_3\ x_4)\ x_5\Big).$$

The translation of a lattice term, say in prefix form, to such a list is clearly a polynomial time procedure.

The following key result for this paper is extracted from Theorem 3.1 of [2].

**Theorem 1.2** (Bloniarz, Hunt III, & Rosenkrantz, 1984)**.** *The set of lattice equations of the form $\bigwedge\bigvee_2 \approx \bigvee\bigwedge\bigvee_{\leq 2}$ that are true of distributive lattices is co-***NP***-hard.*

## 2. Efficient encoding of lattice terms

Let us consider the translation of a lattice term $t$ into any language $\mathcal{L}$ of algebras by using $\mathcal{L}$-terms $r_1(x,y)$, respectively $r_2(x,y)$, to replace occurrences of $\vee$, respectively $\wedge$. We use $\mathrm{T}_{r_1 r_2}(t)$ to denote this translation.

If either $r_1$ or $r_2$ has a repeated variable then such a translation procedure is not polynomial time bounded as the size of the translated terms can grow too fast. [Suppose $r_1(x,y)$ has a repeated variable. Then, for $t = x_1 \vee (x_2 \vee (\cdots (x_{n-1} \vee x_n)\cdots)$, the term $\mathrm{T}_{r_1 r_2}(t)$ has size at least $2^n$.] This was the problem faced, and solved, by Bloniarz, Hunt III, & Rosenkrantz [2] when trying to encode $\mathcal{BA}$ equations of the form $\bigvee\bigwedge_3 \pm var \approx 1$ into the language of rings. (Hunt III & Stearns avoided this problem by translating $\mathcal{BA}$ equations of the form $\bigwedge\bigvee_3 \pm var \approx 0$.) We will expand on their technique to get around the explosive encoding so as to give an efficient procedure for encoding lattice terms. Given a lattice term $t$ it will yield $\mathrm{T}_{r_1 r_2}(t')$ for some lattice term $t'$ which is equivalent (by associativity alone) to $t$. Finding the term $\mathrm{T}_{r_1 r_2}(t')$ will be a polynomial time procedure when restricted to lattice terms whose height is bounded by some constant — this basically generalizes the process used by Bloniarz, Hunt III, & Rosenkrantz [2] (1984), p. 895, on terms in $\bigvee\bigwedge_3$.

### 2.1. The transformations $\tau_{r_i}$.

Given $\mathcal{L}$-terms $r_1(x,y)$ and $r_2(x,y)$ define the transformations $\tau_{r_i}$ from finite sequences $\mathbf{s}$ of lattice terms to $\mathcal{L}$-terms by letting $\tau_{r_i}(\mathbf{s})$ be the single term in the list output by the following program, where $\ell(\mathbf{s})$ is the length of a sequence $\mathbf{s}$:

Let $\mathbf{t} = \mathbf{s}$
LOOP until $\ell(\mathbf{t}) = 1$

Let $\ell = \ell(\mathbf{t})$

$$\text{let } \mathbf{t} \; = \; \begin{cases} (r_i(t_1,t_2),\cdots,r_i(t_{\ell-1},t_\ell)) & \text{for } \quad \ell \text{ even} \\ (r_i(t_1,t_2),\cdots,r_i(t_{\ell-2},t_{\ell-1}),t_\ell) & \text{for } \quad \ell \text{ odd.} \end{cases}$$

ENDLOOP

Let $||\mathbf{t}|| = \sum_{i=1}^{\ell(\mathbf{t})} |t_i|$. Note that $||\mathbf{t}||$ is increasing with every pass through the loop, and the number of passes is bounded by $\log_2 \ell(\mathbf{s}) + 1$. From the easily derived inequality

$$|r_i(t_j, t_{j+1})| \le |r_i(x,y)|(|t_j| + |t_{j+1}|)$$

we see that in a single pass the size of $||\mathbf{t}||$ increases at most by a factor of $2|r_i(x,y)|$. Thus, with $C_i = 2|r_i(x,y)|$, we can conclude that

$$|\tau_{r_i}(\mathbf{s})| \le {C_i}^{\log_2 \ell(\mathbf{s})+1} \cdot ||\mathbf{s}||. \tag{2.1}$$

So the procedure to find $\tau_{r_i}(\mathbf{s})$ is polynomial time.

**2.2. The translation $\tau_{r_1 r_2}$.** We now use these procedures to define a translation $\tau_{r_1 r_2}$ from generalized lattice terms $t^\star$ into $\mathcal{L}$-terms, using $\Box_1$ for $\vee$, $\Box_2$ for $\wedge$:

$$\tau_{r_1 r_2}(x) = x \text{ for } x \text{ a variable}$$

$$\tau_{r_1 r_2}(t_1^\star \,\Box_i \cdots \Box_i\, t_n^\star) = \tau_{r_i}(\tau_{r_1 r_2}(t_1^\star)\,\Box_i \cdots \Box_i\, \tau_{r_1 r_2}(t_n^\star))$$

Observe that if we start with a lattice term $t$ and use $r_1(x,y) = x \vee y$ and $r_2(x,y) = x \wedge y$ then $t' = \tau_{\vee\wedge}(t^\star)$ is a lattice term which is equivalent (using only the associative laws) to $t$. And then for any $\mathcal{L}$-terms $r_1(x,y)$ and $r_2(x,y)$ we have $\tau_{r_1 r_2}(t^\star) = \mathrm{T}_{r_1 r_2}(t')$.

Now fix $\mathcal{L}$-terms $r_1(x,y)$ and $r_2(x,y)$ and choose $C_i$ as in (2.1); and let $C = \max({C_1}^2, {C_2}^2)$.

To measure the complexity of calculating $\tau_{r_1 r_2}(t^\star)$ we introduce the *height* $h(t)$ of a lattice term $t$, defined to be the number of $\bigvee$'s and $\bigwedge$'s in the description of the $(\bigvee, \bigwedge)$ - class to which it belongs. Thus in our example above we have a term of height 2.

**Lemma 2.1.** *For any lattice term $t$ we have*

$$|\tau_{r_1 r_2}(t^\star)| \le |t|^{h(t)\cdot \log_2 C + 1}. \tag{2.2}$$

*Proof.* We proceed by induction on $h(t)$. For $h(t) = 0$ we have $t$ is a variable, thus $\tau_{r_1 r_2}(t^\star) = t$, so (2.2) holds.

For $h(t) > 1$ we assume (2.2) holds for lattice terms of smaller height. Then let

$$t^\star = t_1^\star \,\Box_i \cdots \Box_i\, t_n^\star$$

where $h(t_j) < h(t)$ for $1 \le j \le n$. By definition

$$\tau_{r_1 r_2}(t_1^\star \,\Box_i \cdots \Box_i\, t_n^\star) = \tau_{r_i}(\tau_{r_1 r_2}(t_1^\star)\,\Box_i \cdots \Box_i\, \tau_{r_1 r_2}(t_n^\star))$$

so

$$\begin{array}{rcll}
|\tau_{r_1r_2}(t^\star)| & \leq & C_i^{\log_2 n+1}\displaystyle\sum_{j=1}^{n}|\tau_{r_1r_2}(t_j^\star)| & \text{by (2.1)}\\
& \leq & C_i^{2\log_2|t|}\displaystyle\sum_{j=1}^{n}|t_j|^{h(t_j)\cdot\log_2 C+1} & \text{induction hypothesis}\\
& \leq & C^{\log_2|t|}\displaystyle\sum_{j=1}^{n}|t_j|^{(h(t)-1)\cdot\log_2 C+1} & \\
& \leq & C^{\log_2|t|}\left(\displaystyle\sum_{j=1}^{n}|t_j|\right)^{(h(t)-1)\cdot\log_2 C+1} & \\
& \leq & |t|^{\log_2 C}|t|^{(h(t)-1)\cdot\log_2 C+1} & \\
& \leq & |t|^{h(t)\cdot\log_2 C+1} &
\end{array}$$

□

**Corollary 2.2.** *For lattice terms $t$ with height bounded by some constant, the translation from $t$ to $\tau_{r_1r_2}(t^\star)$ can be carried out in polynomial time.*

**2.3. Application to the complexity of $\mathsf{Th}_{\mathsf{eq}}(\mathcal{K})$.** Now we are ready to give our general setup. With $\vec{x} = (x_1, \cdots, x_n)$ we use $f\vec{x}$ for $(fx_1, \cdots, fx_n)$.

**Proposition 2.3.** *Suppose $\mathcal{K}$ is a class of $\mathcal{L}$-algebras for which there are $\mathcal{L}$-terms $f(x), r_1(x, y), r_2(x, y)$ such that for any lattice equation $s(\vec{x}) \approx t(\vec{x})$*

$$\mathcal{DL} \models s(\vec{x}) \approx t(\vec{x}) \qquad \textit{iff} \qquad \mathcal{K} \models \mathrm{T}_{r_1r_2}(s)(f\vec{x}) \approx \mathrm{T}_{r_1r_2}(t)(f\vec{x}).$$

*Then $\mathsf{Th}_{\mathsf{eq}}(\mathcal{K})$ is co-**NP**-hard. It is co-**NP**-complete if $\mathcal{K}$ is a finite collection of finite algebras.*

*Proof.* Let $s' = \tau_{r_1r_2}(s^\star), t' = \tau_{r_1r_2}(t^\star)$. As $\mathcal{DL}$ satisfies $s \approx s'$ and $t \approx t'$ we have

$$\mathcal{DL} \models s(\vec{x}) \approx t(\vec{x}) \qquad \text{iff} \qquad \mathcal{K} \models \mathrm{T}_{r_1r_2}(s')(f\vec{x}) \approx \mathrm{T}_{r_1r_2}(t')(f\vec{x}).$$

As the set of $s \approx t$ in $\mathsf{Th}_{eq}(\mathcal{DL})$ of the form $\bigwedge\bigvee \approx \bigvee\bigwedge\bigvee$ is co-**NP**-hard by Theorem 1.2, and since the translation

$$\text{from} \qquad s(\vec{x}) \approx t(\vec{x}) \qquad \text{to} \qquad \mathrm{T}_{r_1r_2}(s')(f\vec{x}) \approx \mathrm{T}_{r_1r_2}(t')(f\vec{x})$$

can, for lattice terms of height $\leq 3$, be carried out in polynomial time by Corollary 2.2, it follows that $\mathsf{Th}_{eq}(\mathcal{K})$ is co-**NP**-hard. If $\mathcal{K}$ is finite then $\mathsf{Th}_{eq}(\mathcal{K})$ is clearly co-NP. □

## 3. Using principal congruences for the encoding.

Before delving into the proof of the main result we need to collect some useful definitions and facts. From now on we assume we are working with a fixed nontrivial discriminator variety $\mathcal{V}$, and with a term $t(x, y, z)$ which is a discriminator term on the subdirectly irreducible algebras in $\mathcal{V}$. Define the term $s(x, y, u, v)$ by

$$s(x, y, u, v) = t(t(x, y, u), t(x, y, v), v).$$

Given any algebra $\mathbf{A}$ we know (from Birkhoff's work in the 1930's) that the congruences of $\mathbf{A}$ form a lattice $\mathbf{Con}(\mathbf{A})$. Our strategy is to make heavy use of the nice behavior of congruence lattices in discriminator varieties. $\mathrm{Con}(\mathcal{V})$ denotes the class of lattices $\mathbf{CON}(\mathbf{A})$, for $\mathbf{A} \in \mathcal{V}$. Let $\mathcal{DL}$ denote the variety of distributive lattices.

**Lemma 3.1.** $\mathbf{Con}(\mathcal{V}) \subseteq \mathcal{DL}$.

*Proof.* See Burris & Sankappanavar [3] (1981), p. 165. □

For $\mathbf{A} \in \mathcal{V}$ and $a, b \in A$ let $\Theta(a, b)$ be the *principal congruence* of $\mathbf{A}$ generated by $\langle a, b\rangle$.

**Lemma 3.2.** *For* $\mathbf{A} \in \mathcal{V}$ *and* $a, b, c, d \in A$

(a) $\Theta(a, b) \vee \Theta(c, d) = \Theta(t(a, b, c), t(b, a, d))$
(b) $\Theta(a, b) \wedge \Theta(c, d) = \Theta(s(a, b, c, d), c)$
(c) $\Theta(a, b) = \Theta(c, d)$ *iff*
$t(t(a, b, c)t(a, b, d), t(c, d, a)) = t(t(a, b, d), t(a, b, c)t(c, d, b))$.

*Proof.* The first two are on p. 166 of Burris & Sankappanavar [3] (1981). The third can easily be derived from the other facts on that page. □

**Lemma 3.3.** *Let* $\varepsilon(x_1, \cdots, x_n)$ *be a lattice equation. Then*

$$\mathcal{DL} \models \varepsilon(x_1, \cdots, x_n) \qquad \text{iff} \qquad \mathbf{Con}(\mathcal{V}) \models \varepsilon(\Theta(x_1, y_1), \cdots, \Theta(x_n, y_n)).$$

*Proof.* The direction $(\Longrightarrow)$ is obvious from Lemma 3.2.
For $(\Longleftarrow)$ let $\mathbf{A}$ be a nontrivial simple algebra in $\mathcal{V}$. Then for $a, b \in A$

$$\Theta(a, b) = \begin{cases} \Delta & \text{if} \quad a \neq b \\ \nabla & \text{if} \quad a = b, \end{cases}$$

where $\Delta = \{\langle a, a\rangle : a \in A\}$, and $\nabla = A \times A$. By letting $x_i, y_i$ range over $A$ we see (since $|A| > 1$) that the $\Theta(x_i, y_i)$ can independently take on the values $\Delta, \nabla$. Thus $\varepsilon(x_1, \cdots, x_n)$ holds on the two-element distributive lattice $\langle\{\Delta, \nabla\}, \vee, \wedge\rangle$; so it must hold on all distributive lattices. □

Now we are ready to prove the main result.

**Theorem 3.4.** *Let* $\mathcal{V}$ *be a nontrivial discriminator variety. Then the equational theory of* $\mathcal{V}$ *is co-***NP***-hard. If* $\mathcal{V}$ *is also finitely generated then* $\mathsf{Th}_{eq}(\mathcal{V})$ *is co-***NP***-complete.*

*Proof.* All we need is an efficient way to convert $\varepsilon(\Theta(x_1, y_1), \cdots, \Theta(x_n, y_n))$ into an equation $\varepsilon^\star(x_1, y_1, \cdots, x_n, y_n)$ in the language of $\mathcal{V}$ such that

$$\mathbf{Con}(\mathcal{V}) \models \varepsilon(\Theta(x_1, y_1), \cdots, \Theta(x_n, y_n)) \quad \text{iff} \quad \mathcal{V} \models \varepsilon^\star(x_1, y_1, \cdots, x_n, y_n). \tag{3.1}$$

For then we can use Lemma 3.1. The key idea is to use Lemma 3.2—the first two items of this lemma allow one to eliminate the $\vee$'s and $\wedge$'s; and then the last item gives an equation such that (3.1) holds. Unfortunately a direct application of Lemma 3.2 to $\varepsilon(\Theta(x_1, y_1), \cdots, \Theta(x_n, y_n))$ will sometimes give an exponentially larger $\varepsilon^\star$. To avoid this problem we invoke the results from section 2 to transform $\varepsilon$ into an equivalent $\varepsilon'$ (using just the associative laws) such that applying Lemma 3.2 to $\varepsilon'$ leads to a "small" equation $\varepsilon^\star$ satisfying (3.1). Thus we have, using Lemma 3.1,

$$\mathcal{DL} \models \varepsilon(x_1, \cdots, x_n) \quad \text{iff} \quad \mathcal{V} \models \varepsilon^\star(x_1, y_1, \cdots, x_n, y_n).$$

Consequently $\mathsf{Th}_{eq}(\mathcal{V})$ is co-**NP**-hard, by Lemma 3.3; and by Proposition 1.1, if $\mathcal{V}$ is finitely generated then $\mathsf{Th}_{eq}(\mathcal{V})$ is co-**NP**-complete. $\square$

## References


[1] Stephen L. Bloom and Ralph Tindell, *Varieties of "if-then-else"*. Siam J. Comput. **12** (1983), 677–707.

[2] P.A. Bloniarz, H.B. Hunt III, D.J. Rosenkrantz, *Algebraic structures with hard equivalence and minimization problems.* J. Assoc. for Computing Machinery, **31** (1984), 879–904.

[3] Stanley Burris and H.P. Sankappanavar, *A Course in Universal Algebra.* Springer Verlag, 1981.

[4] Stanley Burris, *Discriminator varieties and symbolic computation.* J. Symbolic Computation **13** (1992), 175–207.

[5] I.N. Herstein, *Noncommutative Rings.* Carus Math. Monographs, Math. Assoc. of America, 1968.

[6] H.B. Hunt III, D.J. Rosenkrantz,& P.A. Bloniarz, *On the computational complexity of algebra on lattices.* Siam J. Comput. **16** (1987), 129–148.

[7] H.B. Hunt III & R.E. Stearns, *The complexity of equivalence for commutative rings.* J. Symbolic Computing **10** (1990), 411–436.

[8] O.G. Kharlampovich and M.V. Sapir, *Algorithmic problems in varieties.* Internat. J. Algebra Comput. **5** (1995), 379–602.

[9] Alan H. Mekler and Evelyn M. Nelson, *Equational bases for if-then-else logic.* Siam J. Comput. **16** (1987), 465–485.



Dept. of Pure Mathematics, University of Waterloo, Waterloo, Ont., Canada N2L 3G1

*E-mail address*: `snburris@thoralf.uwaterloo.ca`