Computers and Universal Algebra: Some Directions

STANLEY BURRIS*

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

(snburris@thoralf.uwaterloo.edu)

The microcomputer revolution has captured the interest of many mathematicians — and Alan Day was certainly a leader in the new wave of computer enthusiasts. Early on he worked out an arrangement with Apple computers; and he was always a strong supporter of their machines. One of his early projects was to create a large database of papers in universal algebra and lattice theory. In the mid 1980's he distributed to his colleagues a program to draw lattices which had menus to check properties such as distributivity and modularity. And the last lecture I heard him give was a lovely talk on relational databases, at McMaster in the Fall of 1989. Unfortunately I am not qualified to give a detailed commentary on Alan's work in computer science ([20], [21]). But I will discuss some aspects of computers and universal algebra which I think Alan would have found interesting.

1 Some Basic Questions

The interest in general algebra and computing goes back at least to Kozen's 1977 paper [38] in which he looks at the complexity of various questions for finitely presented algebras. In his work there is no background equational theory, such as groups, to take into consideration. With this convention he shows that the word problem, the finiteness problem, the triviality problem, and the subalgebra membership problem have polynomial time algorithms,

*Research supported by a grant from NSERC

i.e., the amount of time required is bounded by some polynomial in the size of the presentation. Satisfiability, resp. validity, are NP-complete, resp. co-NP-complete, as in the case of the propositional calculus. (An equation $s(\vec{x}) \approx t(\vec{x})$ is *satisfiable* in an algebra if there is a solution to the equation in the algebra; and it is *valid* in an algebra if it is identically true in the algebra. For the definitions of NP, NP-complete, and co-NP-complete we refer the reader to Garey & Johnson [30].) And he shows the isomorphism problem is isomorphism-complete (see section 3).

Much of the complexity work has focused on finite algebras, rather than finitely presented algebras. A major center of activity developed in Czechoslovakia in the late 1970's. The 1984 survey paper by Kučera & Trnková [39] points out that any first-order property can be checked in polynomial time, e.g., associativity can be checked in $O(n^3)$, where n is the size of the universe. There was considerable interest in finding algorithms to check certain first-order properties (like being a group) which are faster than the obvious brute force check. Some of the results are (again, n is the size of the universe of the algebra):

Abelian group	$O(n^2)$	Tarjan	1972
solvable group	$O(n^2)$	Tarjan	1972
group	$O(n^2)$	Vuillement	1976
ring	$O(n^2)$	Goralčík, Goralčíková, Koubek, Rödl	1981
lattice	$O(n^{5/2})$	Goralčík, Goralčíková, Koubek, Rödl	1981

My first computer algebra project, in the early 1980's, was an attempt to find a ternary discriminator term for the following five-element loop (which we knew to be quasiprimal) on a personal computer with 256 K memory, and no hard disk:

	a	b	c	d	e
a	a	b	c	d	e
b	b	c	d	e	a
c	c	e	b	a	d
d	d	a	e	b	c
e	e	d	a	c	b

This attempt failed, despite trying numerous tricks. So finally I gave the project to my students. The first (and only) student to succeed was David Casperson. The discriminator term he found fills two pages of single-spaced text.

In 1986 an undergraduate student, Wojtek Jastrzebowski, wrote a program which produced the elements of $\mathbf{F}_V(n)$, for $V = V(\mathbf{A})$, $\mathbf{A}$ a finite algebra, after inputting the elements A and the tables for the fundamental operations of $\mathbf{A}$. This was, among other things, quite useful for studying propositional logics. After some soul-searching it was decided a new language was needed for our studies of finite algebras. Wojtek started that summer to design a compact LISP-like language called SLANG, the first version of which he finished the next year as a Masters student at the University of Toronto. When I told Walter Taylor about this language he posed a "usefulness test": to verify that the 59-element countermodel of Gurevič in [34] actually works, namely show that the 59-element algebra of Gurevič satisfies Tarski's High School Identities for the natural numbers with addition, multiplication, exponentiation, and a constant for 1:

$$\begin{array}{cr@{\;\approx\;}l}
(1) & x+y & y+x \\
(2) & x+(y+z) & (x+y)+z \\
(3) & x\cdot 1 & x \\
(4) & x\cdot y & y\cdot x \\
(5) & x\cdot(y\cdot z) & (x\cdot y)\cdot z \\
(6) & x\cdot(y+z) & (x\cdot y)+(x\cdot z) \\
(7) & 1^x & 1 \\
(8) & x^1 & x \\
(9) & x^{y+z} & x^y\cdot x^z \\
(10) & (x\cdot y)^z & x^z\cdot y^z \\
(11) & (x^y)^z & x^{y\cdot z},
\end{array}$$

but not the identity found by Wilkie in 1980 (which is true of the natural numbers):

$$\begin{aligned}
&((1+x)^y+(1+x+x^2)^y)^x\cdot((1+x^3)^x+(1+x^2+x^4)^x)^y \\
&\approx((1+x)^x+(1+x+x^2)^x)^y\cdot((1+x^3)^y+(1+x^2+x^4)^y)^x\,.
\end{aligned}$$

This has led to an ongoing fascination with computer related studies of the finite models of Tarski's High School Identities (see [17], [18]).

In 1987 we began to consider basic facts and questions on complexity results for algebraic structures. To study properties of a finite algebra $\mathbf{A}$ by computer we take as our input data the elements of the universe A and the tables of the fundamental operations. In the following we will be interested in knowing if certain questions about $\mathbf{A}$ can be answered within a polynomial amount of time in the size of our input data.

Recall that McKenzie & Valeriote [42] had proved that a finite algebra $\mathbf{A}$ generates a decidable variety iff $\mathbf{A}$ factors into $\mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \mathbf{A}_3$ with

$V(\mathbf{A}_1)$ a discriminator variety
$V(\mathbf{A}_2)$ a decidable affine variety
$V(\mathbf{A}_3)$ a decidable strongly abelian variety.

We set out to analyze when $\mathbf{A}$ factors into $\mathbf{A}_1 \otimes \mathbf{A}_2 \otimes \mathbf{A}_3$ as above. This led to a number of questions, and we now know the following can be done in polynomial time (given $\mathbf{A}$):

- find $\mathrm{Sg}_{\mathbf{A}}(X)$ (Kozen, 1977)
- find $\Theta_{\mathbf{A}}(a, b)$ (Kozen, 1977)
- determine if $\mathbf{A}$ is subdirectly irreducible (or simple) (Demlová, Demel, Koubek 1979)
- find a subdirect decomposition into subdirectly irreducibles (Jiři Demel, 1982)
- OPEN: Determine if $\mathbf{A}$ is directly indecomposable. Factor $\mathbf{A}$ into directly indecomposables. (Can these be done in polynomial time?)
- find the type set of $\mathbf{A}$ (Berman, Kiss, Pröhle, Szendrei)

  (Hobby shows in [35] that finding the type set of $V(\mathbf{A})$ is NP-hard.)
- determine if $\mathbf{A}$ satisfies TC (TC*) (Berman & McKenzie, 1984)
- find (unique) candidates for $\mathbf{A}_1, \mathbf{A}_2, \mathbf{A}_3$ (one still has to show that $\mathbf{A}_1$ generates a discriminator variety, etc.); or show none exist (Berman)

Our polynomial time progress has been stopped at the following two questions about a finite algebra $\mathbf{A}$:

- Is $V(\mathbf{A})$ a discriminator variety?
- Is $V(\mathbf{A})$ an affine variety?

The apparent bottleneck is:

- Does $\mathbf{A}$ have a Mal'cev term?

The worst-case time bound that we have for this question exceeds $a^{2(a^2-a)r}$, where $a = |A|$ and r is the maximum rank of the operations of $\mathbf{A}$. With such bounds we cannot expect to handle, in general, with current computers, more than three-element algebras; and then we need to limit ourselves to operations of arity ≤ 3.

One is naturally led to inquire if the following have polynomial time algorithms?

- Is $V(\mathbf{A})$ congruence modular?
- Is $V(\mathbf{A})$ congruence distributive?

Related to the question about discriminator varieties we do not know if the following can be answered in polynomial time:

- Is $\mathbf{A}$ quasiprimal?
- Is $\mathbf{A}$ primal?

In testing for $\mathbf{A}$ being primal, Berman pointed out that one can avoid the apparent Mal'cev term bottleneck by using results of Słupecki [46] – still this does not give a polynomial time algorithm (of course such may not exist).

2 Free Spectra

In [3] and [4] Joel Berman has been studying the sizes of the finitely generated free algebras in finitely generated varieties, i.e., the free spectra of such varieties. He has paid special attention to the varieties generated by two- and three-element algebras, and has made substantial use of computer investigations. For a general survey of free spectra see Grätzer & Kisielewicz [33]. Of course the most famous problem in this area is due to Dedekind, namely find an efficiently computable formula for the free spectrum of the variety of distributive lattices (see Quackenbush's survey article [45]). Recently Widemann [50] used a Cray computer to find the 8^{th} Dedekind number.

3 The Isomorphism Problem

The complexity of the isomorphism problem for finite structures is one of the most fascinating open problems in complexity. Any class of finite structures whose isomorphism problem is of the same complexity (modulo polynomial time) as the isomorphism problem for the class of all finite binary structures is said to be *isomorphism-complete*. It is well known that graphs are isomorphism-complete, as are all groupoids, etc.

Kučera and Trnková [40] proved that if V is a variety of unary algebras (of finite type) then either the isomorphism problem for V is polynomial time, or V is isomorphism-complete. Furthermore they gave an algebraic condition, called linearity, on the monoid associated with the variety to determine which case holds. (Valeriote proved, for locally finite varieties, that linearity holds iff the variety V is decidable.) Their result on the complexity of the isomorphism problem for varieties of unary algebras has been extended to strongly Abelian varieties by Valeriote & Willard (unpublished).

Numerous classes of groups and quasigroups have been intensively investigated — see [39]. The variety of lattices is easily seen to be isomorphism-complete. Babai, Klingsberg, and Luks (unpublished) were able to show, using earlier results on graphs, that the isomorphism problem for distributive lattices is polynomial time. And this has been generalized to any finitely generated variety of lattices by Willard (unpublished).

Recently Idziak has informed us that the isomorphism problem for finitely generated arithmetical congruence linear varieties is polynomial time, and that orthomodular lattices are isomorphism complete. Also he and Gorazd have completely classified the complexity of the isomorphism problem for varieties generated by a two element algebra $\mathbf{A}$ — namely the following holds, where $\mathrm{Clo}(\mathbf{A})$ is the clone of $\mathbf{A}$, d is the dual discriminator (we assume $A = \{0,1\}$), and p is ternary addition modulo 2:

$\{d,p\} \subseteq \mathrm{Clo}(\mathbf{A})$	polynomial time
$\{\vee,\wedge\} \subseteq \mathrm{Clo}(\mathbf{A})$	polynomial time
$\mathrm{Clo}(\mathbf{A}) \subseteq \mathrm{Clo}(\{0,1\},+,')$	polynomial time
otherwise	isomorphism complete.

4 The Equivalence Problem

Let $\mathbf{A}$ be an algebra. Then the *term equivalence problem* (TEP) for $\mathbf{A}$ is the complexity of determining if $\mathbf{A} \models s \approx t$, for any terms s and t, i.e., it is the complexity of the equational theory of $\mathbf{A}$. And the *polynomial equivalence problem* (PEP) is defined similarly, replacing "term" by "polynomial". The following table summarizes basic results for finite algebras (the first four assume the algebra is nontrivial):

Boolean algebra	TEP is co-NP-complete	Cook	1971
distributive lattice	”	Bloniarz, Hunt III, Rosenkrantz	1984
finite lattice	”	” ” ”	1987
nonnilpotent ring	”	Burris & Lawrence	1992
nonsolvable groups	”	Lawrence	1992
nilpotent ring	PEP is polynomial time	Hunt & Stearns	1990
nilpotent group	”	Burris & Lawrence	1992
$\mathbf{D}_n$	”	” ”	”

Willard has noted that, as a consequence of results in tame congruence theory, if $\mathbf{A}$ is a finite algebra, $V(\mathbf{A})$ is congruence modular, and $\mathbf{A}$ is not solvable, then the PEP for $\mathbf{A}$ is co-NP-complete. Also he has an example of a finite algebra such that the TEP is polynomial time, and the PEP is co-NP-complete. A study of polynomial time algorithms for the TEP (and the *subcover problem*) for a free lattice is given in Freese's paper [26]. Berman & Blok [6] and Willard (1992, unpublished) have analyzed the equivalence problem for all 2-element algebras $\mathbf{A}$, namely the equivalence problem for $\mathbf{A}$ is polynomial time iff $V(\mathbf{A})$ is not congruence distributive; otherwise it is co-NP-complete. It follows that for 2-element algebras the complexity results for the PEP and TEP coincide.

5 Unification

The concept of unification (and algorithms to find unifiers) has been a pillar of automated theorem proving for the past twenty-five years. It actually concerns straight-forward questions in universal algebra, namely finding solutions of finite sets of equations in a free algebra, determining if every solution comes from a most general solution, and if so, determining how many most general solutions are needed to generate all solutions.

Let V be a variety of algebras, let $\mathbf{F}(n)$ be the n-generated free V-algebra and $\mathbf{F}(\omega)$ the ω-generated free V-algebra. For a finite set of equations Σ in the variables $x_1, \cdots, x_n$ and $\sigma \in \mathrm{Hom}(\mathbf{F}(n), \mathbf{F}(\omega))$ we say σ is a *unifier* of Σ if $\sigma p = \sigma q$ for each $p \approx q \in \Sigma$ (note: we think of p and q as elements of $\mathbf{F}(n)$). Let $U(\Sigma) = \{\sigma \in \mathrm{Hom}(\mathbf{F}(n), \mathbf{F}(\omega)) : \sigma \text{ unifies } \Sigma\}$.

Define a pre-order on $\mathrm{Hom}(\mathbf{F}(n), \mathbf{F}(\omega))$ by: $\sigma_1 \leq \sigma_2$ if there is a $\tau : \mathbf{F}(\omega) \to \mathbf{F}(\omega)$ such that $\sigma_2 = \tau \circ \sigma_1$. For $\sigma_1, \sigma_2 \in U(\Sigma)$ we say σ_1 is *more general than* σ_2 if $\sigma_1 \leq \sigma_2$. Let $\sim$ be the equivalence relation $\leq \cap \geq$. Minimal elements (modulo $\sim$) of $U(\Sigma)$ are called the *most general* unifiers of Σ.

Computer scientists classify the $U(\Sigma)$ into four *unification types* as follows, where the σ_i are understood to be most general unifiers. We assume each successive type excludes the previous types:

- **unitary** means that $U(\Sigma)$ is of the form $\{\sigma : \sigma \geq \sigma_0\}$
- **finitary** means that $U(\Sigma)$ is of the form $\{\sigma : \sigma \geq \sigma_1 \text{ or } \cdots \text{ or } \sigma \geq \sigma_n\}$
- **infinitary** means that $U(\Sigma)$ is of the form $\{\sigma : \sigma \geq \sigma_i, \text{ for some } i < \omega\}$
- **nullary** means that $U(\Sigma)$ is none of the above.

Then one defines the unification type of V to be the worst case among the $U(\Sigma)$, where unitary is 'best' and nullary is 'worst' in our list above. Computer scientists have been interested in finding the unification type of various V, and finding algorithms to produce the most general unifiers. The following table shows part of what is known about the classification of unification types (dates in parentheses mean the results are unpublished):

semigroups	infinitary	Plotkin 1972
commutative semigroups	finitary	Stickel 1975; Livesey & Siekmann 1976
groups	infinitary	Lawrence (1989)
abelian groups	finitary	Lankford 1979
nilpotent class c groups	nullary	Albert & Lawrence (1991)
rings	infinitary	Lawrence (1989)
commutative rings	nullary or infinitary	Burris & Lawrence 1989
lattices	nullary	Willard (1991)
semilattices	finitary	Büttner 1986; Livesey & Siekmann 1976
Boolean algebras	unitary	Büttner & Simonis 1987
primal algebras	unitary	Nipkow 1990
distributive lattices	nullary	Willard (1989)

In Burris [12] it is proved that any discriminator variety has unitary unification type. Albert & Willard (1991, unpublished) have used Post's classification of the clones on a two-element set to determine the unification type of any variety generated by a two-element algebra $\mathbf{A}$. Their results can be summarized as follows, based on the clone of $\mathbf{A}$, $\mathrm{Clo}(\mathbf{A})$, where we say $\mathrm{Clo}(\mathbf{A})$ is CD if the variety generated by $\mathbf{A}$ is congruence distributive:

$\text{Clo}(\mathbf{A}) \subseteq \text{Clo}(\{0,1\},+,')$	unitary
semilattice clones	finitary (one is unitary)
monotone CD clones	nullary
non monotone CD clones	unitary

This major result of course subsumes three of the last four examples in the above table. A number of interesting unification results on algebraic structures associated with logics have been recently obtained by Andrzej Wroński, e.g., every variety of [bounded] Brouwerian semilattices is unitary, every variety of Hilbert algebras is unitary, the variety of bounded Hilbert algebras is not unitary, the variety of p-semilattices is not unitary, every unitary variety of Heyting algebras is contained in KC, and every variety of Heyting algebras in LC is unitary.

6 Term Rewrite Systems

TRS's have been an essential ingredient in automated equational theorem proving for the past 25 years, and there is a substantial literature on this subject (see, e.g., the handbook article of Dershowitz & Jouannaud [24]). TRS's are really just directed equational systems, and one hopes to use them to put terms into normal form. Trevor Evans pioneered the subject in [25] where he shows quasigroups have a normal form TRS. Later Knuth & Bendix [37] showed groups have a normal form TRS — and they made the subject popular with computer scientists by proposing an automated method to convert equational axioms into a normal form TRS. (Their conversion worked for groups.)

Unfortunately equational theories rarely admit a normal form TRS. Plotkin [44] suggested that one should work with ETRS's, namely TRS's modulo some set of equations. Stickel [47] used this idea to find a computer generated equational proof that $x^3 \approx x$ rings are commutative. In [14] we look at discriminator varieties of rings which have a normal form AC TRS (the AC stands for *modulo associativity and commutativity*). George McNulty had asked if lattices have a finite normal form AC TRS, since Whitman's work shows the existence of a natural AC normal form. This has recently been answered in the negative by Freese, Jezek, and Nation in [28]. The last chapter of their book [29] will have a thorough treatment of the computational aspects of lattice theory, including an $O(n^2)$ test for subdirect irreducibility due to Freese.

ACKNOWLEDGEMENTS We are grateful to Joel Berman for his interest in, and contributions to, the first section; and to Ralph Freese for his helpful discussions regarding algorithms for lattices.

References

[1] M.A. Albert and J. Lawrence, *Unification in nilpotent groups.* Preprint, 1992.

[2] D. Benninger and J. Schmid, *Effective subdirect decomposition: a case study.* Theoretical Computer Science **100** (1992), 137–156.

[3] J. Berman, *Algebraic properties of k-valued logics.* Proc. 10^{th} International Symposium on Multiple-Valued Logic, IEEE (1980), 192–204.

[4] J.Berman, *Free spectra of 3-element algebras.* Fourth International Conference on Universal Algebra and Lattice Theory, Proceedings, Puebla 1982; Springer Lecture Notes **1004** (1983), 10–53.

[5] J. Berman, *Finite algebras with large free spectra.* Algebra Universalis **26** (1989), 149–165.

[6] J. Berman and W.J. Blok, *Equational dependencies.* J. Inform. Process. Cybernet. EIK **28** (1992), 215–225.

[7] J. Berman and B. Wolk, *Free lattices in some small varieties.* Algebra Universalis **10** (1980), 269–289.

[8] J. Berman, E. Kiss, P. Pröhle, and A. Szendrei, *The set of types of a finitely generated variety.* To appear in Discrete Math.

[9] J. Berman and R. McKenzie, *Clones satisfying the term condition.* Discrete Math. **52** (1984), 7–29.

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

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

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

[13] S. Burris and J. Lawrence, *Unification in commutative rings is not finitary.* Infor. Proc. Letters **36** (1990), 37–38.

[14] S. Burris and J. Lawrence, *Term rewrite rules for finite fields.* International J. Algebra and Computation **1** (1991), 353–369.

[15] S. Burris and J. Lawrence, *Results on the equivalence problem for finite groups.* Preprint, 1992.

[16] S. Burris and J. Lawrence, *The equivalence problem for finite rings.* J. of Symbolic Computation **15** (1993), 67–71.

[17] S. Burris and S. Lee, *Tarski's High School Identites.* Amer. Math. Monthly **100** (1993), 231–236.

[18] S. Burris and S. Lee, *Small models of the High School Identities.* Internat. J. Algebra and Computation **2** (1992), 139–178.

[19] S. Cook, *The complexity of theorem proving procedures.* Proc. Third Annual ACM Symposium on the Theory of Computing, 1971, 151–158.

[20] A. Day, *The lattice theory of functional dependencies and normal decompositions.* International J. Algebra and Computation (to appear).

[21] A. Day, *A lattice interpretation of database dependencies.* "Semantics of Programming Languages and Model Theory" (M. Droste and Yu. Gurevich, eds), Gordon and Breach, London, (to appear).

[22] J. Demel, *Fast algorithms for finding a subdirect decomposition and interesting congruences of finite algebras.* Kybernetika (Prague) **18** (1982), 121–130.

[23] M. Demlová, J. Demel, V. Koubek, *Several algorithms for finite algebras.* Proc. of FCT 79, Akademie-Verlag Berlin 1979, 99–104.

[24] N. Dershowitz and J.-P. Jouannaud, *Rewrite systems.* Chap. 6 of the Handbook of Theoretical Computer Science, ed. J. van Leeuven, Elsevier Science Publishers, 1990, 245–320.

[25] T. Evans, *On multiplicative systems defined by generators and relations.* J. Proc. Cambridge Philos. Soc. **47** (1951), 637–649.

[26] R.S. Freese, *Free lattice algorithms.* Order **3** (1987), 331–34.

[27] R.S. Freese, *Finitely presented lattices: canonical forms and the covering relation.* Trans. Amer. Math. Soc. **312** (1989), 841–860.

[28] R.S. Freese, J. Jezek, and J.B. Nation, *Term rewrite systems for lattice theory.* Preprint, 1992.

[29] R.S. Freese, J. Jezek, and J.B. Nation, "Free Lattices", in preparation.

[30] M.R. Garey and D.S. Johnson, "Computers and Intractability: a Guide to the Theory of NP-Completeness." W.H. Freeman & Co., 1979.

[31] P. Goralčík, A. Goralčíková, and V. Koubek, *Testing of properties of finite algebras.* Proc. of ICALP 80, Lecture Notes in Comp. Sci. **85**, Springer Verlag, 1980, 273–281.

[32] P. Goralčík, A. Goralčíková, V. Koubek, and V. Rödl, *Fast recognition of rings and lattices.* Proc. of FCT 81, Lecture Notes in Comp. Sci. **117**, Springer Verlag, 1981, 137–145.

[33] G. Grätzer and A. Kisielewicz, *A survey of some open problems on p_n-sequences and free spectra of algebras and varieties.* Preprint, 1991.

[34] R. Gurevič, *Equational theory of positive numbers with exponentiation.* Proc. Amer. Math. Soc. **94** (1985), 135–141.

[35] D. Hobby, *Finding type sets is NP-hard.* Internat. J. Algebra and Computation **1** (1991), 437–444.

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

[37] D.E. Knuth and P.B. Bendix, *Some simple word problems in universal algebras.* in J. Leech, ed., Computational Problems in Abstract Algebra, Pergamon Press, Oxford, 1970, 263–297.

[38] D. Kozen, *Complexity of finitely presented algebras.* Proc. 9^{th} Symposium STOC (1977), 164–177.

[39] L. Kučera and Trnková, *The computational complexity of some problems of universal algebra.* Universal Algebra and its Links with Logic, Algebra, Combinatorics, and Computer Science. Proc. "25. Arbeitstagung über Allgemeine Algebra", Darmstadt 1983, P. Burmeister et al. (eds.), Heldermann Verlag 1984, 216–229.

[40] L. Kučera and Trnková, *Isomorphism testing of unary algebras.* Siam. J. of Computing **17** (1988), 673–686.

[41] D. Lankford, *A unification algorithm for Abelian group theory.* Internal Report MTP-1, Louisiana Tech. Univ., Ruston, 1979.

[42] R. McKenzie and M. Valeriote, *Decidable Varieties.* Birkhauser, 1987.

[43] T. Nipkow, *Unification in primal algebras, their powers, and their varieties.* J. Assoc. Comp. Mach. **37** (1990), 742–776.

[44] G. Plotkin, *Building in equational theories.* Mach. Intelligence **7**, Halsted-Press, New York, 1972, 227–261.

[45] R.W. Quackenbush, *Dedekind's prolem.* Order **2** (1986), 415–417.

[46] J. Słupecki, *Completeness criteria for systems of many-valued propositional calculus.* Comptes rendus des Séances de la Société des Sciences et des Letters de Varsovie Cl. II. **32** (1939), 102–109. English transl. Studia Logica **30** (1972), 153–157.

[47] M.E. Stickel, *A case study of theorem proving by the Knuth-Bendix method: discovering that $x^3 \approx x$ implies ring commutativity.* 7^{th} Internat. Conf. on Automated Deduction, Napa, California, Lecture Notes in Comp. Sci. **170** (1984), 248–258.

[48] R.E. Tarjan, *Determining whether a groupoid is a group.* Information Processing Letters **1** (1972), 120–124.

[49] J. Vuillement, *Comment verifier si une boucle est un groupe.* Technical Report, Université de Paris-Sud, 1976, 1–12.

[50] D. Wiedemann, *A computation of the eighth Dedekind number.* Order **8** (1991), 5–6.