# EQUATIONAL LOGIC AND ABSTRACT ALGEBRA*

*Taje I. Ramsamujh*
*Florida International University*
*Mathematics Department*

*ABSTRACT*

Equational logic is a formalization of the deductive methods encountered in studying the set of all equations that can be derived from a given fixed set of equations. So it is naturally associated with abstract algebraic structures. The equations involved are interpreted as being true for all the variables involved and so are best thought of as identities. In complexity, equational logic sits somewhere between propositional and first-order logic. And even though it may appear simple at first sight, many of the problems are very interesting and non-trivial. Several are actually quite difficult and some are still open. In this exposition our goal is to introduce equational logic in an informal way as a bridge from the propositional logic to the first-order logic and contrast it with them.

---

* This is an expanded version of a talk presented at the MAA Florida Section Annual Meeting at FGCU, March 2nd-3rd, 2001.

## 1. Introduction.

Equational logic is often referred to as *universal algebra* because of its natural association with abstract algebraic structures but this is not the view we shall take here. We shall view it as a logic and accordingly begin by giving a general outline of what is a logic. There is a large body of work on equational logic and we refer the reader to the very comprehensive survey by Taylor [1979], even though it may be a little dated. Our goal is to show how equational logic can be use to motivate the study of first-order logic and model theory. Along the way we shall present a few interesting problems as they arise, including the resolution of the famous *Robbins conjecture* by an automated theorem prover.

A *logic* consists of an *alphabet* of symbols, a syntax and a semantics. It is similar in many ways to a written language and can, in fact, be considered a written language. The symbols in the alphabet are called *letters* and the *syntax* specifies the way in which these letters are used to construct *formulas*. The syntax has to be such there is an algorithm to determine in a finite number of steps whether or not an arbitrary finite string of letters is a formula. The set of all formulas is called the *language* of the logic. The *semantics* specifies the intended meaning of the formulas by *interpreting* the language in a structure. A *structure* consists of a set A (called the *domain* of the structure) together with certain relations and functions on A of various arities. The arity is the number of variables involved in the relation or function.

A *sentence* is a formula with no free variables. In propositional and equational logic all formulas are sentences. In first-order logic, $(\exists x)(\forall x)(x+y = x-y)$ is a sentence but $(\exists x)(x.y = x+y)$ is not. Let G be a set of sentences and ? be a given sentence. We say that ? is a *logical consequence* of G if ? is true in all structures in which all the sentences of G are true.

A *theory* is a set T of sentences, in some underlying logic, which is closed under logical consequences (i.e., if ? is a logical consequence of T, them ? must be in T). If the underlying logic is equational logic (propositional or first-order logic), then we will call the theory an *equational theory* (*propositional* and *first-order theory*, respectively).

There are two basic ways of specifying theories. If G is any set of sentences, we can get a theory by letting T be the set of all logical consequences of G and we will write this as T = Conseq(G). We can also get a theory by letting T be the set of all sentences that are true in a fixed structure A or in every member of a whole class of structures $\{A:A\in C\}$. We will write this as T=Th(A) or as $T=Th(\{A:A\in C\})$

There are three basic questions about theories.

(a) Is the theory T algorithmically decidable ?
(b) Is the theory T axiomatizable ?
(c) Is the theory T consistent ?

A set G of formulas is *algorithmically decidable* if there is an algorithm which can determine whether or not an arbitrary sentence s is in G. This tells us what it means for a theory to be algorithmically decidable. We say that a theory T is *axiomatizable* if there is a algorithmically decidable set G of formulas such that T = Conseq(G). Usually we have a formal deductive system for the underlying logic and augment it with G. If the underlying logic is sufficiently nice (and all three of the logics we will discuss, are) we get a formal deductive system $S_T$ such that $s\in T$ if and only if s is derivable in $S_T$.

A *formal deductive system (F.D.S.)* S consists of the language of a logic, an algorithmically decidable set A(S) of sentences called the axioms, and a finite set R(S) of rules of inferences. The *axioms* are certain carefully selected formulas which are usually obviously true in some intended interpretation of the language of the logic. A *rule of inference* is a rule that is usually true in all intended interpretations and specifies that one formula (called the *conclusion*) can be deduced from a finite set of formulas (called the *hypotheses*). A sentence s is *derivable* in S if there is a sequence $s_1, s_2, \ldots, s_n = s$ such that each $s_i$ is either an axiom or is deducible by a rule of inference from previous $s_j$'s in this sequence.

The axioms of A(S) are usually divided into two parts - the *logical axioms* LA(S) (which axiomatizes the underlying logic and does not change with the theory) and the *proper axioms* PA(S) (which consists of the remaining axioms of A(S) and varies with the theory).

The set R(S) of rules of inference is part of the underlying logic because it does not change with the theory.

Once we know that a theory is axiomatizable, the second question can be further refined. Is T *finitely axiomatizable* (i.e., is there a F.D.S. for T with a finite set of proper axioms)? Is T *n-axiomatizable* (i.e., is there a F.D.S. for T with a set of n proper axioms)? This latter question only makes sense for equational theories because any propositional or first-order theory which is finitely axiomatizable will be 1-axiomatizable. We just have to take a conjunction of the finitely many proper axioms to get one proper axiom.

A theory T is *consistent* if it does not consist of all possible sentences in the language of T. In the propositional and first-order logic a theory T is usually said to be "*in-consistent*" if it contains both a sentence and its negation. It can be shown that when this happens T must contain all possible sentences, because T is closed under logical consequences. A theory T is said to be *maximal* if it is consistent and there is no consistent theory which properly contains T. In the propositional and first-order logic, a theory T is usually said to be "*maximal*" if for each sentence s, either $s \in T$ or $\neg s \in T$. It can be shown that when this happens there is no consistent theory which properly contains T.

**2. Three logics**.

We shall first introduce *propositional logic* (PL). The alphabet of PL consists of:

| | |
|---|---|
| (a) *connectives*: | $\perp$ (falsum), $\rightarrow$ (conditional) |
| (b) *auxiliary symbols*: | (, ) (parentheses) |
| (c) *relation symbols*: | $P_{0,i}$ ($k \in I$) |

Here I is an indexing set which is usually taken to be the set of natural numbers N, but it can be anything including the empty set. Strictly speaking there are several propositional logics because the set of relation symbols may vary - but this does not change things very much. The set of relation symbols is referred to as the *proper part* of the alphabet. *Falsum* is a 0-ary connective and the *conditional* is a binary

connective. For each $i \in I$, $P_{0,i}$ is a *0-ary relation symbol*. The parentheses are just used for punctuation.

The formulas of PL are defined recursively as follows:

1. $\perp$ and each $P_{0,i}$ are formulas
2. if a and ß are formulas, then so is $(a \rightarrow ß)$.
3. a is a formula if and only if it can be obtained from 1 by a finite number of applications of step 2.

We shall refer to the formulas of PL as complex propositions. The other connectives can be introduced as abbreviations for easier comprehension and readability.

| | |
|---|---|
| *negation*: | $(a \rightarrow \perp)$ abbreviates $(\neg a)$ |
| *disjunction*: | $(\neg a \rightarrow ß)$ abbreviates $(a \vee ß)$ |
| *conjunction*: | $\neg(\neg a \vee \neg ß)$ abbreviates $(a \wedge ß)$ |
| *biconditional*: | $(a \rightarrow ß) \wedge (ß \rightarrow a)$ abbreviates $(a \leftrightarrow ß)$ |

The intended meaning of is "$\perp$" is the 0-ary connective which is always false. For all practical purposes "$\perp$" can be taken to be the constant proposition which is always false. "$\neg\perp$" will then be the constant proposition which is always true and is called *verum* and abbreviated as "_". $(a \rightarrow ß)$ has the usual meaning of "a implies ß" - it is false if and only if a is true and ß is false.

The 0-ary relation symbols $P_{0,i}$ are to be interpreted as 0-ary relations. An n-ary relation on a structure with domain A is just a subset of $A^n$. Here $A^n$ is the set of all sequences of length n. $A^0$ consists of a single element - the *empty sequence* ?, i.e., $A^0 = \{?\}$. So a 0-ary relation on A will either be $\varnothing$ or $\{?\}$. We interpret $\varnothing$ as the relation being false and $\{?\}$ as it being true. So a 0-ary relation is either true or false when interpreted over a structure - in other words it acts just as the usual proposition variable. We have formulated propositional logic in this way so that it becomes a sub-logic of first-order logic.

Let $T_{PL}$ be the set of all complex propositions that are logical consequences of the empty set $\varnothing$ of complex propositions. The member of $T_{PL}$ are usually called *tautologies.* It has long be known that

$T_{PL}$ is algorithmically decidable. The usual *truth-table method* provides a simple algorithm to determine whether or not a formula is in $T_{PL}$. The same is true if we replace $\varnothing$ by a *finite* set of complex propositions H to get the propositional theory $T_{PL}(H)$. A natural question is what happens when H is *infinite*? Here, as the reader may expect, the answer is in the negative.

There are also several elegant formal deductive systems for $T_{PL}$. One F.D.S. has three axioms and two rules of inference - substitution and modus ponens. Another has three axiom schemas (and thus an infinite number of axioms) and only modus ponens as the rule of inference. (See Mendelson [1997] for these and others.) If we write _s to mean that s is a logical consequence of the empty set of sentences and _s to mean that s is derivable in one of the F.D.S. above, then it can be shown that _s if and only if _s. This is called the *completeness theorem* for propositional logic. So, anyway, $T_{PL}$ is axiomatizable. It is also consistent (the sentence $\perp$ is not in $T_{PL}$) but *not* maximal (neither $P_{0,1}$ nor $\neg P_{0,1}$ is in $T_{PL}$).

Next we will discuss *first-order logic* (FL). The alphabet of FL consists of:

| | | |
|---|---|---|
| (a) *connectives*: | | $\perp, \rightarrow$ |
| (b) *universal quantifier* | | $\forall$ |
| (c) *equality symbol:* | | $=$ |
| (d) *auxiliary symbols*: | | (, ), "," |
| (e) *individual variables*: | $x_k$ | $(k \in N)$ |
| (f) *relation symbols*: | | $P_{n,i}$ $(n,i \in I_n)$ |
| (g) *function symbols*: | | $f_{n,j}$ $(n,j \in J_n)$. |

Here $I_n$ and $J_n$ are indexing sets which are usually taken to be N, but they can be anything including the empty set. Strictly speaking there are several first-order logics because the set of relation symbols and function symbols, which are referred to as the *proper part* of the alphabet, may vary - but this does not change things very much. We allow the relation symbols and function symbols to be of any arity $n \geq 0$. The *comma* "," is an auxiliary symbol to help with punctuation.

First we define the terms of FL. A *term* is defined recursively as follows:

1. For each $k \in N$ and each $j \in J_0$, $x_k$ and $f_{0,j}$ are terms.
2. If $n \geq 1$ and $t_1,...,t_n$ are terms then so is $f_{n,j}(t_1,...,t_n)$.
3. t is a term if and only if it can be obtained from 1 by a finite number of applications of step 2.

The formulas of FL are defined recursively as follows:

1. $\perp$ and each $P_{0,i}$ are formulas.
2. If $t_1$ and $t_2$ are terms then $(t_1=t_2)$ is a formula.
3. If $n \geq 1$ and $t_1,...,t_n$ are terms, then $P_{n,i}(t_1,...,t_n)$ is a formula.
4. If a and ß are formulas, then so are $(a \rightarrow ß)$ and $((\forall x_k)a)$.
5. a is a formula if and only if it can be obtained from 1, 2 & 3 by a finite number of applications of step 4.

The existential quantifier and the other connectives from PL can be introduced as abbreviations for easier comprehension and readability.

*existential quantifier*: $(\exists a)$ abbreviates $\neg((\forall x_k)(\neg a))$

The 0-ary function symbols $f_{0,j}$ are to be interpreted as 0-ary functions. A 0-ary function f on A is just a function from $A^0=\{?\}$ to A. Since f is completely determined by f(?), f is basically an individual constant from A. The intended meaning of $(\forall x_k)a$ when interpreted in a structure with domain A is "for all $x_k$ in A, a holds". If the variable $x_k$ does not occur in a, then $(\forall x_k)a$ and a have the same meaning. In first-order logic the equality symbols always has to be interpreted as the identity relation on A.

Let $T_{FL}$ be the set of all sentences of FL that are logical consequences of the empty set of sentences. Then it is well known that $T_{FL}$ is *not* algorithmically decidable. This is essentially what makes the task of finding "proofs" in ordinary mathematics rather difficult for beginning students. There are however several elegant formal deductive systems for $T_{FL}$. One F.D.S. has seven axiom schemas and two rules of inference - generalization and modus ponens. (See Mendelson [1997]

for more details.) With the F.D.S. mentioned above, it can be shown that _s if and only if _s. This is called the *Godel completeness theorem* for first-order logic. Anyway, $T_{FL}$ is axiomatizable. It is also consistent (the sentence $\perp$ is not in $T_{FL}$) but *not* complete (neither $P_{0,1}$ nor $\neg P_{0,1}$ is in $T_{FL}$).

Finally we shall discuss *equational logic* (EL). The alphabet of EL consists of:

| | | |
|---|---|---|
| (a) *equality symbol:* | | = |
| (b) *auxiliary symbols*: | | (, ), "," |
| (c) *individual variables*: | $x_k$ | $(k \in N)$ |
| (d) *function symbols*: | | $f_{n,j}$ $(n,j \in J_n)$ |

The symbols here have all been mentioned above in connection with FL. The *terms* of EL are defined in the same way as in FL. The formulas of FL are defined simply as the set of all expressions of the form $(t_1=t_2)$, where $t_1$ and $t_2$ are terms. We shall refer to the formulas of EL as *equations* or *identities*. The only difference between EL and FL is in the way in which we interpret the equality symbol. We consider an equation of EL to be true in a structure with domain A if it is true for all values of the variables involved in the equation. We can view an equation of EL as formulas of FL by prefacing it by universal quantifiers with each of the variable involved in the equation.

Let $T_{EL}$ be the set of all equations that are logical consequences of the empty set $\varnothing$ of equations. Then $T_{EL}$ is algorithmically decidable. (In fact $T_{EL}$ is really a trivial theory. It consists of all equations of the form (t=t) where t is an arbitrary term of EL.) The same thing is *not* always true, however, if we replace $\varnothing$ by a *finite* set of equations H to get the equational theory $T_{EL}(H)$. It has been shown by Tarski [1953] that there is a finite set H of equations such that $T_{EL}(H)$ is *not* algorithmically decidable. In the next section we will present a formal deductive system for equational theories and discuss this matter further later. In passing we would like to mention that $T_{EL}$ is consistent ($x_0=x_1$ is not in $T_{EL}$) and that it is not maximal, as long as the proper part of the alphabet is non-trivial. It is, in fact, *minimal* - this means that it is contained in all other equational theories.

## 3. A formal deductive system for equational theories.

The only logical axiom is: A1. $x_0=x_0$ (*identity axiom*)
The proper axioms will depend on the theory in question.

The rules of inference are as follows:
R1. From s=t, deduce t=s. (*symmetry rule*)
R2. From r=s and s=t, deduce r=t. (*transitivity rule*)
R3. From $s_1=t_1, \ldots, s_n=t_n$ deduce,
$f_{n,k}(s_1,...,s_n) = f_{n,k}(t_1,...,t_n)$. (*replacement rule*)
R4. From $r(x_1,...,x_n) = s(x_1,...,x_n)$ deduce,
$r(t_1,...,t_n) = s(t_1,...,t_n)$ (*substitution rule*)

We can replace the two rules R1 and R2 by one rule R5.
R5. From r=s and s=t, deduce t=r. (*circularity rule*)

Let us look at some algebraic structures to see how this F.D.S. actually works. The proper part of the alphabet of the *theory of groups* of $T_{GR}$ consists of:

(a) a binary function symbol: . (*multiplication*)
(b) a unary function symbol: ' (*inverse*)
(c) a 0-ary function symbol: e (*identity element*).

The proper axioms of $T_{GR}$ are:

G1. (x.y).z = x.(y.z) (*associativity*)
G2. x.e = x (*right identity*)
G3. x.x' = e (*right inverse*)

Here we have taken x, y, z as variable instead of $x_1$, $x_2$, $x_3$ to simplify the expressions. A *group* is any structure $\langle A,.,',e\rangle$ in which G1-G3 are satisfied. A *semi-group* is any structure $\langle A,.\rangle$ in which G1 is satisfied. A *groupoid* is just a structure $\langle A,.\rangle$ in which no equations are required to be satisfied. The theory of groups $T_{GR}$ is the set of all equations that are logical consequences of $G_{GR}$ = {G1,G2,G3}.

It can be shown that an equation s is a logical consequence of $G_{GR}$ if and only if s is derivable in our F.D.S above supplemented with

$G_{GR}$ as proper axioms. This is really saying that $G_{GR}$_s if and only if $G_{GR}$_s. This result is true if we replace $G_{GR}$ by an arbitrary set G of equations and it is known as the *Birkhoff completeness theorem* for equational logic. G is often referred to as a *base* of the theory Conseq(G).

We will illustrate the concept of a deduction by considering the following questions:

Q1. Is x'.x = x true in all groups ?
Q2. Is e.x = x true in all groups ?

As the reader knows very well, both questions can be answered in the affirmative. We will provide deductions of these two equations from $G_{GR}$. Below is a deduction of x'.x = e.

| | | |
|---|---|---|
| 1. | x'.x = (x'.x).e | G2 |
| 2. | x'.x = (x'.x).[(x'.x).(x'.x)'] | G3 |
| 3. | x'.x = [(x'.x).(x'.x)].(x'.x)' | G1 |
| 4. | x'.x = [{(x'.x).x'}.x].(x'.x)' | G1 |
| 5. | x'.x = [{x'.(x.x')}.x].(x'.x)' | G1 |
| 6. | x'.x = [{x'.e}.x].(x'.x)' | G3 |
| 7. | x'.x = [x'.x].(x'.x)' | G2 |
| 8. | x'.x = e | G1 |

In the above deduction we introduced [] and {} for ease of readability. Using this deduction, we next give a deduction of e.x = x.

| | | |
|---|---|---|
| 1. | e.x = (x.x').x | G3 |
| 2. | e.x = x.(x'.x) | G1 |
| 3. | e.x = x.e | previous deduction |
| 4. | e.x = x | G2 |

A careful reader would have noticed that we formulated the equational theory of groups in an alphabet which allowed us to express the three group axioms as identities. If the proper part of the alphabet consisted only of the binary function symbol "." we would not have been able to do this. The vast majority of algebraic structures allows us

to study their equational theories, although a little unorthodoxy may be involved in a few cases. These structures include semi-groups, quasi-groups, abelian groups, rings, rings with identity, commutative rings, lattices and Boolean algebras. By viewing scalar multiplication by each element of a field or ring with identity as a separate unary function, the same can be said for vector spaces over a fixed field and unital modules over a fixed ring with identity. The theory of fields or division rings cannot, however, be cast as equational theories - unless you really go against normal practices.

**4. Finitely axiomatizable theories.**

An equational theory is said to be *finitely axiomatizable* if there is a *finite* set G of equations such that T = Conseq(G). For the rest of this article all the theories mentioned will be equational theories with finitely many function symbols. So all of our theories will have a countable base. Finitely axiomatizable theories are often referred to as *finitely based* because the *base* G is finite. There are, of course, theories that are not finitely based - but most naturally formulated theories are finitely based. For any finite structure S, the theory Th(S) is obviously finitely based. And if a theory T, such as group theory, is defined by T = Conseq(G) where G is finite - it is *de facto* finitely based.

An interesting example of a non-finitely based theory is Th(G) where G={0,1,2} is the *Lyndon groupoid* with the binary operation defined below:

| . | 0 | 1 | 2 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 |
| 2 | 0 | 2 | 2 |

Since G is not finitely based this means that there is no finite set of equations which can generate all the equations that are satisfied by this groupoid. Another more natural example is Th(P) where P is the *Perkins semi-group* which consists of the six 2x2 matrices shown below under matrix multiplication.

| | | | | |
|---|---|---|---|---|
| 0 0<br>0 0 | 1 0 | 1 0 | 0 1 | 0 0 |
| 0 0<br>0 1 | 0 1 | 0 0 | 0 0 | 1 0 |

A final very interesting example is Th(HA) where HA is the "high school algebra" structure $\langle ?,+,.,\uparrow\rangle$ with $\uparrow$ being exponentiation.

A theory is said to be *one-based* if it has a base which consists of one equation. It was shown by McKenzie [1970] that the theory of lattices was is one-based - his original proof yielded an equation with 34 variables and of length about 300,000. (The author was tempted to put an exclamation mark after the 300,000 - but that might make the length appear to be even bigger than it really was!) Padmanabhan has reduced it to a formula involving 7 variable and of length about 300. Gratzer, McKenzie & Tarski also showed that the theory of Boolean algebras was one-based (Gratzer [1971]). Neither of these proofs were easy and the single equations were not particularly revealing.

All two element groupoids can be divided into seven classes such that any two groupoid in the same class are isomorphic or dual-isomorphic. Five of the classes have theories that are one-based and the remaining two, whose representatives are $\langle\{0,1\},\vee\rangle$ and $\langle\{0,1\},\rightarrow\rangle$, have been shown by Potts [1965] to be *two-based*. Here 0 and 1 are considered as the truth values "false" and "true" and $\vee$ and $\rightarrow$ have their usual meaning from propositional logic.

A base is said to be *irredundant* if no proper subset of it is a base. It is a little surprising that a single theory can have irredundant bases with different cardinalities - so there is no concept such as the "dimension" of a theory. (But not very surprising because the cyclic group of 6 elements can have a set consisting of one generator or a set consisting of two generators.) It can be shown, however, that every irredundant base of a finitely based theory must have finite cardinality. There is a beautiful result of Tarski [1968] that says that the possible values of the cardinalities of the irredundant bases of a finitely based *non-trivial* theory is either a finite or an infinite interval of positive integers, i.e., either [m,n] or [m,? ). Here ? is the first infinite cardinal and is better known as aleph-null. The *trivial theory* consisting of all

tautological identities has the empty set as a base and thus an irredundant base of cardinality 0.

For non-finitely based theories there are two possibilities: there is either an irredundant base of cardinality ? or there is no irredundant base. In the latter case this means that every base has a proper infinite subset which is also a base. No mathematically interesting theory without irredundant bases have been found so far and it would be interesting to know if there is an infinite group G such that Th(G) has no irredundant base. If a theory has an infinite irredundant base G it will automatically have $2^{?}$ many different sub-theories because there are $2^{?}$ infinite subsets of G and each subset will produce a different sub-theory. The same may not be true for theories with no irredundant base. The most we can say at this point is that it will have at least ? many different sub-theories.

**5. Decision problems**.

A *decision problem* can be characterized as a problem in which there are an infinite number of inputs and for each input there is a YES or NO answer. Also each input must be finitely specifiable so there will be a denumerable number of inputs. A typical decision problem is the *primality problem*:
Given an arbitrary number n, can we tell if it is prime? Here the inputs are natural numbers and are usually presented in base 10 as a finite string of digits. We say that a decision problem is *algorithmically solvable* if there is a *single algorithm* which when given the input computes the correct answer in a finite number of steps. It is well known that the primality problem is algorithmically solvable. We just have to check if n is divisible by some prime $\leq \lceil \sqrt{n} \rceil$.

The *halting problem* is as follows: Given an arbitrary Turing machine, can we tell if it will halt when started on the blank tape ? It is the standard example of a decision problem which is not algorithmically solvable and many decisions problems are shown to be algorithmically unsolvable by showing that an affirmative answer would result in the halting problem being algorithmically solvable.

We now turn our attention to decision problems in equational logic. But before doing this we recall that Tarski [1953] showed that there is a finite set H of equations such that the theory $T_H$ = Conseq(H) is not algorithmically decidable. This means that the decision problem for $T_H$ is not algorithmically solvable, i.e., there is no algorithm which can tell us if an arbitrary equation s is in $T_H$ or not. (By the way the decision problems for most equational theories, such as group theory and ring theory, are algorithmically solvable.) Below are some examples of algorithmically unsolvable decision problems. In everything that follows G will be an *arbitrary finite set of equations.*

U1. Is the theory Conseq(G) consistent ?
U2. Is the theory Conseq(G) maximal ?
U3. Is the theory Conseq(G) one-based ?
U4. Is the theory Conseq(G) algorithmically decidable?
U5. Is G a base for the theory of groups?
U6. Is G a base for the theory of Boolean algebras

Problem U1 is equivalent to asking if the equation x=y is derivable from G, because the equation x=y is a base for the inconsistent theory which consists of all possible equations.

Not all the problems of these types are algorithmically unsolvable. Below are some other examples of algorithmically solvable decision problems.

S1. For an arbitrary finite structure A, is Th(A) maximal?
S2. Is G a base for the theory of commutative groupoids?
S3. Is the theory Conseq(G+G1) consistent ?

Here G1 is the associative rule from group theory. A commutative groupoid is one which satisfies x.y = y.x .

In connection with problem U6, we would like to mention the Robbins conjecture. A more complete discussion is given in Cipra [1999] and McCune and Padmanabhan [1996]. *Robbins conjecture* asks if the following three equations form a base for the theory of Boolean algebras:

1. $x \vee y = y \vee x$ (*commutativity*)

2. $x \vee (y \_ z) = (x \vee y) \vee z$ (*associativity*)
3. $((x \vee y)' \_ (x \vee y')')' = x$ (*Robbins axiom*)

This problem was open for 63 years until McCune, building on the work of others, recently obtained a solution from his automated theorem prover (a computer program called EQP). Note the algorithmic unsolvability of U6 only means that there is no algorithm which can solve all problems of the format of Robbins conjecture - it did not mean that Robbins conjecture could not be settled. What is remarkable here is that this seemingly simple problem defied the efforts of some of the best mathematicians but succumbed to the brute force of a computer. But that is not the complete story - there was a human being who wrote the program with the aid of experts in equational logic such as S. Burris and R. Padmanabhan. And here I abruptly end my tale for it has gotten much too long!

**6. References.**


Birkhoff, Garret

[1935] *On the structure of abstract algebras*, Proc. Cambr. Philos. Soc. 31 (1935), 433-454.

Cipra, Barry

[1999] *As easy as EQP* in: *What's happening in the Mathematical Sciences*, Vol. 4 (1998-99), 59-66, American Math. Soc., Providence RI.

Gratzer, G.

[1971] *Lattice Theory, First concepts and distributive lattices*, H.M. Freeman, San Francisco.

McCune, W. & Padmanabhan, R.

[1996] *Automated Deduction in Equational Logic and Cubic Curves*, Springer-Verlag, Berlin.

McKenzie, R.

[1970] *Equational bases for lattice theories*, Math. Scand., 27 (1970), 24-38.

Mendelson, Elliott

[1997] *Introduction to Mathematical Logic*, 4th edition, Chapman & Hall, New York.

Potts, D. H.

[1965] *Axioms for semi-lattices*, Canadian Math. Bulletin 8 (1965), 519.

Tarski, Alfred

[1953] [*Abstracts*], J. Symbolic Logic 18 (1953), 188-189

[1968] *Equational logic and equational theories of algebra*, 275-288 in: H.A. Schmidt, ed., *Contributions to Mathematical Logic*, North Holland, Amsterdam.

Taylor, Walter

[1979] *Equational Logic*, Houston J. Math., Survey 1979.