THE EFFICACY OF RUE RESOLUTION
EXPERIMENTAL RESULTS AND HEURISTIC THEORY

Vincent J. DIgrIcoli

Courant Institute of Mathematical Sciences
New York University and Hofstra University
166-11 17th Road, Whitestone, New York 11357

ABSTRACT

We present and analyse experimental results in the first extensive use of a theorem prover based on Resolution by Unification and Equality. Implicit use is made of equality axioms by the Inference rules RUE and NRF to achieve incisive refutations for E-unsatlsflability. Since a primary issue in automated deduction is the efficacy of convergence to proof, we describe in detail the heuristics which were used to obtain proofs. A comparative tabulation with the results of McCharen, Overbeek and Wos, who used unification resolution, shows sharply reduced cumulative unification counts.

1. Introduction

A primary reason for the failure of binary resolution relates to the handling of equality, a predicate which plays a central role in most axiomatic systems. The use of the equality axioms In the input clause set leads to very long proofs which are unnecessarily cumbersome and paramodulation which leads to much shorter proofs presents problems of heuristic search for the proper paramodulants which experimentors have not been able to surmount.

In [8,9] the author presented the theory of resolution by unification and equality in which the axioms of equality are incorporated into the definition of resolution. In contrast to paramodulatlon, RUE resolution does lend itself to heuristics which converge to refutations and it is the purpose of this paper to carefully study this issue.

We will first give a synopsis of the basic theorems and definitions of RUE resolution as presented in [9] and then proceed to the discussion of experiments and heuristic search procedures.

(1.1) A <u>disagreement set of</u> a <u>pair of terms</u> (t_1,t_2) is defined in the following manner:

(1) If t_1,t_2 sre identical the empty set is the sole disagreement set.

(2) If t_1,t_2 are not identical, the set of one element the pair (t_1,t_2) is the <u>origin disagreement set</u>.

(3) If t_1,t_2 have the form

$$t_1= f(a_1,\ldots,a_k),\ t_2= f(b_1,\ldots,b_k)$$

then the set of pairs of corresponding arguments which are not identical is the disagreement set at the topmost argument level or more simply the <u>topmost disagreement</u> of t_1,t_2.

(4) If D is a disagreement set of t_1,t_2, then D' obtained by replacing any member of D by the elements of one of its disagreement sets is also a disagreement set of t_1,t_2.

In the simple example:

$$t_1 = f(a,\ g(b,h(c))$$

$$t_2 = f(a',g(b',h(c'))$$

besides the origin disagreement, there are the disagreement sets:

$$D_1 = \{a:a',\ g(b,h(c)):\ g(b',h(c'))\}$$

$$D_2 = \{a:a',\ b:b',\ h(c):h(c')\}$$

$$D_3 = \{a:a',\ b:b',\ c:c'\}\ .$$

The purpose of this definition is to define all possible ways of proving $t_1 = t_2$, i.e. we can prove $t_1 = t_2$ by proving equality in every pair of any one disagreement set. An input clause set, for example, may imply equality in D_1 but not in D_2 or D_3. Or it may most directly prove $t_1 = t_2$ by proving equality in D_3.

In the sequence of derivable disagreement sets from topmost to bottommost, there is upward implication of equality, $D_{i+1} \rightarrow D_i$, but not downward implication, $D_i \not\rightarrow D_{i+1}$.

(1.2) We proceed to define a <u>disagreement set of complementary literals</u>:

$$P(s_1,\ldots,s_n)\ ,\qquad \bar{P}(t_1,\ldots,t_n)$$

as the union of disagreement sets:

$$D = \bigcup_{i=1,n} D_i$$

where D_i is a disagreement set of (s_i,t_i).

We see immediately that:

$$P(s_1,\ldots,s_n) \wedge \bar{P}(t_1,\ldots,t_n) \rightarrow D$$

where D now represents the disjunction of inequalities specified by a disagreement set of $P,\bar{P}$, and furthermore, that:

$$f(a_1,\ldots,a_k) \neq f(b_1,\ldots,b_k) \rightarrow D$$

where D is the disjunction of inequalities specified by a disagreement set of $f(a_1,\ldots,a_k)$, $f(b_1,\ldots,b_k)$.

We may now define our two rules of inference:

(1.3) <u>The RUE Rule of Inference</u>

"Given the clauses

$$A \vee P(s_1,\ldots,s_n) \quad \text{and} \quad B \vee \bar{P}(t_1,\ldots,t_n)$$

and a substitution σ, the RUE resolvent of σ applied to these clauses is

$$\sigma A \vee \sigma B \vee D$$

where D is the disjunction of inequalities specified by a disagreement set of the complementary literals σP, $\sigma\bar{P}$."

(1.4) <u>The NRF Rule of Inference</u>
<u>(the negative reflexive function rule)</u>

"Given $A \vee t_1 \neq t_2$ and a substitution σ, the NRF resolvent of σ applied to this clause is $\sigma A \vee D$, where D is the disjunction of inequalities specified by a disagreement set of σt_1, σt_2."

We have for example:

$$P(f(a,b),c) \wedge \bar{P}(f(a',b'),c')$$
$$\rightarrow f(a,b) \neq f(a',b') \quad \vee \quad c \neq c'$$

and

$$P(f(a,b),c) \wedge \bar{P}(f(a',b'),c')$$
$$\rightarrow a \neq a' \quad \vee \quad b \neq b' \quad \vee \quad c \neq c'$$

which are obviously not equivalent deduction*. If we instead use the equality axioms with unification resolution, each of the above one step deductions becomes an elongated sequence of 3 and 6 steps respectively.

The two important issues which remain are the selection of substitution and disagreement set when we resolve complementary literals, for which there is the theorem:

(1.5) <u>Completeness</u> <u>Theorem</u> (Strong Form):

"A set of clauses is E-unsatisfiable if and only if there is an RUE-NRF deduction of the empty clause from S, in which we:

(1) choose as σ, the RUE-NRF unifier,

(2) choose as D, the topmost viable disagreement set,

(3) satisfy the equality restriction when resolving complementary equality literals."

The reader will find in (9] a detailed treatment of the notion of viability which subtends the definitions of the above unifying substitution and also the equality restriction. It suffices to say. here that we are to use the MGPU (the most general partial unifier) of P, P and this is a qualified form of the NGU of standard resolution, relaxed to permit irreducible disagreements.

The viability criterion enables us to suppress deductions which are recognized as provably irrelevant to a refutation and the equality restriction enables us to avoid useless variants of the same deduction path.

The above is a very cursory description of a quite substantial body of theory presented in [9].

2. An RUE <u>Refutation in Group Theory</u>:

Below we present an RUE refutation which proves that a group is commutative if x*x - e. The axioms of group theory are given in (4.3) of this paper. The style of proof Is similar to that introduced by Harrison and Rubin in generalized resolution [7].

In formulating substitutions we are scanning complementary literals from left to right, skipping over Irreducible disagreements, and furthermore substituting first at the topmost argument level and then at lower levels. This is evident in the first two steps of the refutation.

We are also being appropriately selective in choosing a disagreement set, sometimes using the topmost and sometimes the bottommost according to a criterion we will develop later in (5.2).

This 9 step proof becomes a 22 step proof when we use the equality axioms with unification resolution. The RUE theorem prover deduced the above refutation in 2.5 seconds using a total of 570 unifications. Typically a 9 step RUE proof will represent a fairly sophisticated axiomatic derivation which humans do not easily arrive at.

(2.1)

		Substitution
ab ≠ ba		
	xe = x	ba/x
(ba)e ≠ ab		
	x(yz) = (xy)z	ba/x, b/z
yb ≠ e v (ba)y ≠ a		
	xx = e	y/x, b/y
(ba)b ≠ a		
	ex = x	a/x
ea ≠ (ba)b		
	(xy)z = x(yz)	a/z, ba/x
(ba)y ≠ e v ya ≠ b		
	xx = e	ba/x, ba/y
(ba)a ≠ b		
	(xy)z = x(yz)	b/x, a/y, a/z
b(aa) ≠ b		
	xe = x	b/x
e ≠ aa		
	e = xx	a/x
empty clause		

3. A Comparative Analysis

In evaluating the performance of an automatic theorem prover, It is always important to compare machine performance against human performance and in doing so to avoid extremes. We should not expect to capture in a computer algorithm the creative, analytical genius of an exceptional mathematician, rather we should be content to compare against astute human performance. We should require a theorem prover to prove theorems which astute humans find difficult to deduce. Since the reader will always consider himself an astute human, he can put the issue subjectively to test.

At this level of performance, automated deduction has important applications in many fields, including robotics, the science of intelligent machines, and also data base technology which should be more than sophisticated collection schemes for explicitly stored data.

Since 1965 when resolution was introduced, we have accumulated substantial experience with machine deduction by resolution. As recently as 1976, the Computer Transactions of the IEEE contained two important articles of major implementations of resolution theorem provers.

In the Maryland system of Wilson and Minker [5], the authors conclude with a very negative assessment of the efficacy of resolution:

> "Considering that the average proof attempt using the best overall proof procedure required 42 CPU seconds, our results are discouraging. Even though most of the problems were relatively simple for a human to solve, the total CPU time required for the study cost more than $30,000. Clearly such a theorem proving system is far from practical.
> None of the inference systems tested enabled more than a marginal improvement in the overall power of unrestricted binary resolution. ... We have been unable to determine the problem and proof procedure characteristics which either enable problems to be solved or prevent them from being solved."

In contrast in the Argonne system of HcCharen, Overbeek and Wos [6], the authors present 63 experiments ranging from trivial to quite difficult theorems, proven in diverse fields, together with some theorems in which their system failed. Their experiments represent one of the most successful attempts in automatic deduction. They presented for each experiment the input clause set used as well as a systematic tabulation of the amount of computation required. This paper in singular fashion established a benchmark for other researchers to compare with and we have here used it as a basis of evaluation of RUE.

We have conducted 17 experiments taken from HOW and dealing with Ring Theory, Group Theory and Boolean Algebra. The heuristic procedures developed from these experiments have, thus far, proven effective. We state the comparative results across 17 experiments between MOW and RUE in Table I.

Table 1. Comparative Tabulation

		RUE		MOW	
	Length of RUE Proof	Total Unif.	Time Sec.	Total Unif.	Time Sec.
Boolean Algebra					
B1 $\overline{0} = 1$	(2)	42	0.1	26,702	16.2
B2 x+1 = 1	(5)	606	1.2	46,137	28,5
B3 x*0 = 0	(5)	978	1.9	46,371	27.5
B4 x+xy = x	(8)	746	1.4		
B5 x(x+y) = x	(8)	1086	2.0		
B4 ∧ B5				286,902	57.0 (5)
B6 x+x = x	(5)	851	1.6		
B7 x*x = x	(5)	756	1.4		
B6 ∧ B7				105,839	60.6 (6)
Ring Theory					
R1 x*0 = 0	(5)	114	0.2	12,328	9.83
R2 0*x = x	(5)	199	0.4	NPR	
R3 xy = (-x)(-y)	(13)	10,850	25.9	94,031	45.08
Group Theory					
G1 $x^2=e$ + yz=zy	(9)	570	2.5	830	1.0
G2 inverse of x is unique	(5)	338	0.5	504	1.0
G3 left identity is right identity	(10)	1114	21.0 (7)	901	2.0
G4 x has a right inverse	(7)	401	2.8	474	5.43
G5 $(x^{-1})^{-1} = x$	(5)	200	0.9	NPR	
G6 $(xy)^{-1} = y^{-1}x^{-1}$	(9)	1932	26.0 (7)	NPR	
G7 $x^3 = e + y^{-1} = y^2$	(5)	586	2.4	NPR	

Notes on Table 1:

(1) The RUE theorem prover was exceptionally superior throughout.the ten boolean and ring theory experiments, and in group theory both theorem provers achieved impressive results.

(2) The RUE theorem prover was implemented in the PL/1 programming language and run on an Amdahl computer. The MOW theorem prover was implemented in Assembly language and run on the IBM System 370/195. It is difficult to compare runtimes in diverse hardware and software environments, and we suggest total unifications as a valid measure of comparison.

(3) Since MOW used a complete set of equality axioms in the input clause set of all these experiments and since they do not tabulate any paramodulatlon counts, it seems that they made little successful use of paramodulatlon even though their paper states that this technique was incorporated in their system.

(4) NPR: no published reeult without implying that MOW felled on these theorems.

(5) MOW ran BA and B5 as a single theorem

(x + xy - x) A x(x + y) • x

which negated becomes a+ab=a v a(a+b)f*a
In the Input clause set, and It Is evident that the refutation of ~(B4 A B5) IS simply the concatenation of the refutations for ~B4 and ~B5 Individually taken and as derived by RUE.

(6) The same holds for ~(B6 A B7) whose proof is the concatenation of the RUE refutations for ~B6 and ~B7 individually taken.

(7) In proving G3 and G6, the RUE theorem prover compared each new resolvent against each resolvent already in the refutation search tree in order to suppress duplicates and the overhead led to the very slow times 21 and 26 seconds. However, this duplicate removal can be implemented much more efficiently to reduce the runtimes considerably. For the most part, duplicate removal was not applied in the other RUE theorems of Table 1.

4. Experiments

The reader may wish to assess the difficulty of the above theorems by attempting their proof:

(4.1) Given the Boolean axioms

1. $x+y = y+x$
2. $xy = yx$
3. $x+0 = x$
4. $x*1 = x$
5. $x+\bar{x} = 1$
6. $x*x = 0$
7. $x(y+z) = xy+xz$
8. $x+yz = (x+y)(x+z)$

prove the theorems:

B1. $\tilde{0} = 1$
B2. $x+1 = 1$
B3. $x*0 = 0$
B4. $x+xy = x$
B5. $x(x+y) = x$
B6. $x+x = x$
B7. $x*x = x$

(4.2) Given the axioms in Ring Theory

1. $x+y = y+x$
2. $x+0 = x$
3. $x+(-x) = 0$
4. $x(y+z) = xy+xz$
5. $(x+y)z = xz+yz$
6. $x+(y+z) = (x+y)+z$
7. $x(yz) = (xy)z$
8. $x=y \quad v \quad x+z \neq y+z$

prove the theorems:

R1. $x*0 = 0$
R2. $0*x = 0$
R3. $xy = (-x)(-y)$

(4.3) Given the axioms of Group Theory

1. $ex = x$
2. $xe = x$
3. $x^{-1}x = e$
4. $xx^{-1} = e$
5. $x(yz) = (xy)z$

prove the theorems:

G1. $x*x = e \rightarrow yz = zy$
(a group is commutative if $x^2 = e$)

G2. $(xz=e \wedge zx=e) \wedge (yz=e \wedge zy=e) \rightarrow x=y$
(the inverse of z is unique)

G5. $(x^{-1})^{-1} = x$

G6. $(xy)^{-1} = y^{-1}x^{-1}$

G7. $x^3 = e \rightarrow y^{-1} = y^2$

Furthermore, with the reduced axioms set:

(4.31) 1. $ex = x$ 2. $x^{-1}x = e$ 3. $x(yz) = (xy)z$

prove the theorems:

G3. $xe = x$

G4. $xx^{-1} = e$

With RUE a computer proved all of the above 17 theorems In 92 seconds which may be compared with the reader's own achievement. The brevity and Incislveness of the computer deduced RUE proofs [10] can also be compared with the reader's own humanly deduced proofs.

The above represent first experiments with RUE automated deduction, wherein the longest theorem proven is 13 RUE steps (27 steps If using the equality axioms), and we will now outline the heuristics which were derived and employed in these first experiments.

5. Heuristic Principles Applied

The following are the primary heuristic principles which were developed during experimentation:

(1) Heuristic ordering by degree of unification

(2) Selection of the loweat level disagreement set not containing an irreducible literal

(3) Heuristic substitution selection

(A) Complexity bounds relating to:

(a) argument nesting
(b) number of distinct variables in a resolvent
(c) number of occurrences of the same constant or function symbol in a clause
(d) maximum number of literals in a clause
(e) maximum character length of a clause

(5) Purging redundancies by subsumptlon

(6) Demodulation

(7) Frequency bounds for the use of individual axioms in a refutation path and bounds on the consecutive use of the same axiom.

All of the above principles are syntactic in nature and apply generlcally to experiments performed. (1) through (3) ae RUE specific but the remaining principles have commonly been used by resolution theorem provers.

(5.1) Heurletlc Ordering by Degree of Unification

If we wish to erase the literal $t_1 \neq t_2$, we measure the relevancy of an axiom $a_1 = a_2$ for this task by computing the degree of unification between complementary literals in the following manner:

```
(1) Apply the MGPU to complementary literals
    to obtain $\sigma t_1 \neq \sigma t_2$, $\sigma a_1 = \sigma a_2$

(2) Set $w=0$ (unification weight)

(3) For $i=1,2$
    if $\sigma t_i$ matches $\sigma a_i$ identically
    then $w = w+50$
    else if $\sigma a_i$, $\sigma t_i$ are the same
    function, say

    $\sigma a_i = f(b_1,b_2)$,  $\sigma t_i = f(c_1,c_2)$,

    then $w = w+20$ and, furthermore, $w = w+15$
    for each matching pair of corresponding
    arguments.
```

This is a simple scheme of matching which computea a weight of 100 when oa_1«oa_2 erases ctifot? and a weight of 0 when the complementary literals do not satisfy the equality restriction. There is also an intermediate scoring between these extremes.

For example, in the refutation of ab=ba stated in (2.1) which proves that a group Is commutative if x*x • e, we used the input clause set:

S:	1.	$ex = x$	2.	$x = ex$
	3.	$xe = x$	4.	$x = xe$
	5.	$x^{-1}x = e$	6.	$e = x^{-1}x$
	7.	$xx^{-1} = e$	8.	$e = xx^{-1}$
	9.	$x(yz) = (xy)z$	10.	$(xy)z = x(yz)$
	11.	$xx = e$ (negated theorem)		
	12.	$e = xx$		
	13.	$ab \neq ba$		

Note that we have added to the input clause set the symmetry variant of each axiom which enablea ua in this experiment to build in symmetry without introducing the axiom *x4y* v y-x. This technique was also used by NOW.

Table 2 shows the unification acoring which took place for aucceaalve resolvents in the refutation of ab=ba. We place in parentheaea axloma having a common score and * above the axiom actually used in the refutation. We refer to lxioms by their index in the input clause set.

Table 2. (Axiom Unification Scoring)

$ab \neq ba$	:	$(1,2,\overset{*}{3},4,9,10)$-70, $(5,6,7,8,11,12)$-35
$(ba)e \neq ab$	:	(3)-85, $(1,2,4,\overset{*}{9},10)$-70, $(5,6,7,8,11,12)$-35
$yb \neq e$	:	$(5,\overset{*}{11})$-100, $(1,3,7)$-85, $(9,10)$-60, $(2,4)$-50, $(6,8,12)$-0
$(ba)b \neq a$	:	$(\overset{*}{1},3)$-70, $(9,10)$-60, $(2,4)$-50, $(5,7,11)$-35, $(6,8,12)$-0
$ea \neq (ba)b$	:	$(1,2,3,4,9,\overset{*}{10})$-70, $(5,6,7,8,11,12)$-35
$(ba)y \neq e$	:	$(7,\overset{*}{11})$-100, $(1,3,5)$-85, $(9,10)$-60, $(2,4)$-50, $(6,8,12)$-0
$(ba)a \neq b$	:	$(1,3)$-70, $(9,\overset{*}{10})$-60, $(2,4)$-50, $(5,7,11)$-35, $(6,8,12)$-0
$b(aa) \neq b$	:	$(\overset{*}{3})$-85, (1)-70, $(9,10)$-60, $(2,4)$-50, $(5,7,11)$-35, $(6,8,12)$-0
$e \neq aa$	:	$(1\overset{*}{2})$-100, $(2,4,6,8)$-70, $(9,10)$-60, $(1,2)$-50, $(5,7,11)$-0

The relative ranking of the refutation axlon In each acorlng la given by the sequence: 3,5,2,1,6,2,4,1,1. The refutation axioms average to a relative position of 2.78 among 12 candidate axioms.

Furthermore, If we write the refutation as i welghted-axloro sequence:

					(axiom weight)				
$ab \neq ba$	70	70	100	70	70	100	60	85	100
	3	9	11	1	10	11	10	3	12
					(axiom index)				

we see that the minimum weight Is 60 and that wc may disregard an axiom scoring below 60 In the refutation search.

We may now state our first principle ol heurlstlcally ordering the expansion of th< refutation search tree:

1. Apply axioms to a negative literal In the order of higher degree of unflcalon first and set a lower limit <u>SDWMIN</u> below which we suppress or postpone the application of an axiom (search directive weight minimum)

2. Furthermore, among axioms which qualify fo application, select the first <u>SPLIM</u> candidates (search directive limit).

In reviewing all the refutations in group theory, we found the SDWMIN 50 or 60 applied. Though the completeness theory specifies that all axioms for which w > 0 must be used to preserve completeness, heurlstlcally w >- SO specifies a subspace of search where we should expect to find the refutation.

Furthermore, SDLIM specifies an inner-subspace of the latter which is even richer in expectation. In the Boolean Algebra and Ring Theory experiments, SDLIM 3 or A was successfully used.

(5.2) Selectinq the Lowest Level Disaqreement Set Not Containinq an Irreducible Literal

Typically in adding the negated*theorem to the input clause set, we Introduce skolem constants and when It is evident that these constants are in effect arbitrary constants with respect to the input axioms, then we can conclude that Inequalities on skolem constants like a=b are irreducible, I.e. we cannot deduce a-b from the axiom set.

For example, in group theory the negated theorem ab=ba Introduces skolem constants a,b, which with respect to the axioms:

ex-x, xe«x, x~*x"e, xx^-e, x(y«)"(xy)z

are arbitrary constants and we cannot prove from these axioms that a-b. We should never generate an inequality af*b in a disagreement set during a refutation search. Furthermore, inequalities like (a~^{1}a)b-e which demodulate to Irreducible literals, are also irreducible and can never appear In a refutation.

This leads us to the following heuristic rule:

"In an RUE deduction, the disagreement

set likely to be required by a refuta-

tion, Is the lowest level disagreement

set not containing an irreducible literal."

In fact across 17 experiments containing 109 refutation steps, the above selection was always correct and what is more important led to proofs which did not require NRF.

The above rule heurlstlcally complements the strict rule specified by the completeness theory where we state that we must select the topmost viable disagreement set, which then requires us to descend to lower levels by NRF If necessary.

(5,3) Heuristic Substitution Selection

In the completeness analysis In [9], we establish that substitutions are to be performed only in variables at the first argument level of predicates in RUE and only In variables at the first argument level of the outermost function In NRF. This qualification of the MGPU It required for completeness. It leads, however, to proofs with extensive use of the NRF rule, yielding much longer refutations requiring deeper search trees. These same proofs can always be expressed in shorter form without NRF steps, by using Immediate substitutions at lower argument levels*

Intuitively It Is desirable to use an unrestricted MGPU, achieving a maximum degree of unification so that Inequalities In a disagreement set are reduced In number.

It In fact occurs In all 109 refutation steps In 17 experiments performed that maximum unification as described below Is the refutation substitution:

1. First subsltute In a left-to-rlght scan only In variables at the first argument level.
2. Then In a second left-to-rlght scan, substitute In variables at all lower argument levels.

Let us now apply both the notion of maximum unification and Irreducible literals to our refutation of abffba In group theory. From the constants

$$(a,a^{-1},b,b^{-1},e)$$

we define as Irreducible literals:

$$(a\neq a^{-1},\ a\neq b,\ a\neq b^{-1},\ a\neq e,$$

$$a^{-1}\neq b,\ a^{-1}\neq b^{-1},\ a^{-1}\neq e,$$

$$b\neq b^{-1},\ b\neq e,$$

$$.b^{-1}\neq e\ )$$

and augment this set with any literal which demodulates to one of Its members, using the demodulating substitutions

$$(\ ex \rightarrow x,\ xe \rightarrow x,\ x^{-1}x \rightarrow e,\ xx^{-1} \rightarrow e\ ).$$

If In searching for a refutation of ab=ba, we heur1stleally choose the maximum unification MGPU and the lowest level disagreement set not containing an Irreducible literal , we obtain the refutation we have already stated In (2.1). This proof which is specified by the heuristic theory is 9 steps compared to the 17 step refutation specified by the completeness theory which contains 8 additional NRF steps.

6. Completeness vs. Heuristic Theory

There s a polarity between completeness and heuristic theory, the latter tends to sacrifice completeness for the sake of efficiency which is crucial, and the former in giving the highest priority to the preservation of completeness tends to be seriously inefficient. It is important, however, to elaborate both approaches, and we have dona this In [9] for coapleteness and hare for heuristics.

7. Longest Experiment Performed

Let us now examine the longest refutation in Table 1, containing 13 RUE steps, which proves in ring theory that xy - (-x)(-y). The RUE theoren prover deduced this refutation in less than 26 seconds employing 10,850 unifications in its heuristic search. The same proof with the equality ixloms would be a 27 step refutation.

			Substitution
(7.1)	ab ≠ (-a)(-b)		
		x=y v x+z≠y+	ab/x, (-a)(-b)/y
	ab+z ≠ (-a)(-b)+z		
		xu+yu=(x+y)u	a/x, b/u, yb/z
	(a+y)b ≠ (-a)(-b)+yb		
		x=u v x+z≠u+	(a+y)b/x, (-a)(-b)/u
	(a+y)b+z ≠ ((-a)(-b)+yb)+z		
		uw+vw=(u+v)w	a+y/u, b/w, vb/z
	((a+y)+v)b ≠ ((-a)(-b)+yb)+vb		
		x=0+x	((a+y)+v)b/x
	0 ≠ (-a)(-b)+yb v (a+y)+v ≠ v		
		x(u+z)=xu+xz	-a/x, -b/u, -a/y, b/z
	-a(-b+b) ≠ 0 v (a+(-a))+w ≠ v		
		x=y v x+z≠y+	-a(-b+b)/x, 0/y
	-a(-b+b)+z ≠ 0+z v (a+(-a))+v ≠ v		
		x=0+x	-a(-b+b)+z/x
	-a(-b+b)+z ≠ z v a+(-a)+v ≠ v		
		ux+uy=u(x+y)	-a/u, -b+b/x, -ay/z
	(-b+b)+y ≠ y v a+(-a)+v ≠ v		
		0+x=x	y/x
	0 ≠ -b+b v a+(-a)+v ≠ v		
		0= -x+x	b/x
	(a+(-a))+v ≠ v		
		0+x=x	v/x
	0 ≠ a+(-a)		
		0= x+(-x)	a/x
	empty clause		

The above refutation can be summarised as follows: "From the negated theorem $ab \neq (-a)(-b)$ and the axiom $x \neq y$ v $x+z \neq y+z$, form the construct

$$ab + y\underline{b + vb} \neq (-a)(-b) + y\underline{b + vb}$$

which factors to the form

$$(a + y + v)b \neq (-a)(-b) + yb + vb \ .$$

Substitute $-a/y$ and interpret as follows:

$$(\underline{a + (-a))} + v)b \neq -a(-b+b) + vb$$

$$(0 + v)b \neq \quad " \quad "$$

$$vb \neq \quad " \quad "$$

and we only need to prove $-a(-b+b) = 0$ which the refutation does as follows:

$$
\begin{aligned}
-a(-b+b) &\neq 0 \\
-a(-b+b) + z &\neq 0 + z \\
-a(-b+b) + z &\neq z
\end{aligned}
$$

substitute $-ay/z$ and factor

$$
\begin{aligned}
-a(-b + b + y) &\neq -ay \\
(-b + b + y) &\neq y \\
0 + y &\neq y \\
y &\neq y \qquad \text{q.e.d.}
\end{aligned}
$$

8. Conclusion

The above gives us a measure of the logical complexity of a 13 step RUE proof. We expect the current RUE theorem provr to be consistently effective in finding proofs of length 15, and we are now proceeding to attempt to prove very long theorems in the neighborhood of 40 steps, such as the following proven by MOW:

"In group theory, using the extended axiom set, prove

$$x^3 = e \rightarrow h(h(x,y),y) = e$$

$$\text{where } h(x,y) = xyx^{-1}y^{-1}.$$

Using the equality axioms and the additional lemmas

$$(x^{-1})^{-1} = x \text{ and } e^{-1} = e \ ,$$

MOW proved this theorem in 54 seconds using 184,955 unifications. Wos in [3] presented a 38 step paramodulation proof (humanly deduced) for this theorem without using additional lemmas. The RUE proof should be of the same length.

If automatic deduction can be brought to be consistently successful in proving theorems such as the above, this will be a landmark achievement.

In the end there is an event which can serve to signal the maturity of automatic theorem proving, namely the appearance of controlled competitions (as in chess) between theorem provers and finally between machine and astute human.

References

[1] Robinson, J. A., "A Machine Oriented Logic Based on the Resolution Principle", JACM 12, 23-41, 1965.

[2] Wos, Lawrence, Robinson, G. A., Carson, D.F., "The Concept of Demodulation In Theorem Proving", JACM, Vol. 14, No. 4, Oct. 1967.

[3] Robinson, G. and Wos, L., "Paramodulatlon and Theorem Proving in First Order Theories with Equality", Machine Ingelllgence, Vol.4, 1969.

[4] Morris, J., "E-resolution: An Extension of Resolution to Include the Equality Relation", Proc. IJCAI, 1969.

[5] Wilson, G. A., Mlnker, J., "Resolution, Refinements, and Search Strategies: A Comparative Study", IEEE Transactions on Computers, Vol. C-25, No. 8, Aug. 1976.

[6] McCharen, Overbeek and Wos, "Problems and Experiments for and with Automated Theorem Proving Programs", IEEE Transactions on Computers, Vol. C-25, No. 8, Aug. 1976.

[7] Harrison, M., Rubin, N., "Another Generalization of Resolution", JACM, Vol. 25, No. 3, July 1978.

[8] Dlgricoll, V. J., "Resolution by Unification and Equality", Proceedings of the 4th Workshop on Automated Deduction, Univ. of Texas, Feb. 1979.

[9] Dlgricoll, V. J., "Automatic Deduction and Equality", Proceedings of the Oct. 1979 Annual Conference of the ACM, 240-250.

[10] Dlgricoll, V. J., "Resolution by Unification and Equality", Technical Report, Dept. of Computer Science, Courant Institute, New York University, 251 Mercer St., New York, N. Y. 10012, Sept. 1981. (The moat complete reference for RUE theory and experiments)•

[11] Wos, L., Overbeek, R., Henschen, L., "Hyperparamodulation: A Refinement of Paramodulatlon", 5th Conference on Automated Deduction, July, 1980.