ATP-19
September 1974

CANONICAL FORM OF A SET OF MODELS *

by

Robert Anderson **

and

Vesko Marinov ***

September 1974 ATP-19

* This research was supported by grant GM-15769-05 from the National Institute of Health

** Present address: University of Houston, Department of Computer Sciences

*** Present address: Stanford University, IMSSS

Abstract

The canonical form of a set of models $\mathcal{M}$ is defined to be the set S' of all clauses deducible by resolution from a set of ground clauses S, which has $\mathcal{M}$ as its only models, after all tautologies and subsumed clauses are deleted. It is shown that S' is the same for any S having $\mathcal{M}$ as its only models (i.e., $S' = CF(\mathcal{M})$). Furthermore, it is shown that all literals in the intersection of $\mathcal{M}$ occur as unit clauses in $CF(\mathcal{M})$. An algorithm for producing $CF(\mathcal{M})$ and its implementation in UT-LISP are given.

1. Introduction

In doing proofs by resolution one reduces the question of validity to showing that a set of clauses is unsatisfiable. If it is the case that a set of clauses is satisfiable, then it will have some models (i.e., interpretations in which all the clauses are true). In general, there are many different sets of clauses which have the same set of interpretations as their only models.

When a set S consists entirely of ground clauses then the set of clauses deducible from S by resolution (i.e., $\mathcal{R}^{\infty}(S)$) is finite. From $\mathcal{R}^{\infty}(S)$ we can obtain the Reduced $(\mathcal{R}^{\infty}(S))$ by deleting all tautologies and subsumed clauses. It turns out that if we are given a set of models $\mathcal{M}$ then any set S which has $\mathcal{M}$ as its only models also has the same Reduced $(\mathcal{R}^{\infty}(S))$. We call this the canonical form of $\mathcal{M}$ - $CF(\mathcal{M})$. The above is established in Section 2 of this report.

In Section 3 it is shown that all literals which are in the intersection of a set of models $\mathcal{M}$ occur as unit clauses in $CF(\mathcal{M})$. Section 4 gives an algorithm for producing $CF(\mathcal{M})$. This algorithm has been implemented in UT-LISP on CDC 6600. Some examples of sets of models and their canonical forms produced by this program are shown in Appendix I. The program and a brief working documentation are given in Appendices II and III, respectively.

2. Canonical Forms

First we shall prove a few theorems leading to the definition of canonical forms.

Corollary 1. If S_1 and S_2 are any two sets of ground clauses which have identically the same models, then given any clause $C_1 \in \mathcal{R}^\infty(S_1)$ there exist clauses $C_1^0 \in \mathcal{R}^\infty(S_1)$ and $C_2^0 \in \mathcal{R}^\infty(S_2)$ such that $C_2^0 = C_1^0 \subset C_1$.

Proof. Suppose we have such S_1, S_2, and C_1. Then by Theorem 2 there is a clause $C_2 \in \mathcal{R}^\infty(S_2)$ such that $C_2 \subset C_1$. Now using Theorem 2 again there must be a $C_1' \in \mathcal{R}^\infty(S_1)$ such that $C_1' \subset C_2 \subset C_1$. Thus using Theorem 2 still again we get a $C_2'' \in \mathcal{R}^\infty(S_2)$ such that $C_2'' \subset C_2' \subset C_2 \subset C_1$. This chain cannot go on indefinitely since C_1 only has a finite number of literals in it. Hence there are some clauses in the chain being produced, call them C_1^0 and C_2^0, such that $C_1^0 \in \mathcal{R}^\infty(S_2)$ and $C_2^0 = C_1^0 \subset C_1$.

Definition 1. If S is any set of clauses then the reduced set of S, denoted Reduced (S), is the set of clauses obtained from S by deleting any tautologies and any clauses which are subsumed by other clauses in S.

Thus given any clause $C \in$ Reduced (S) one can be sure that there is no other clause $C' \in$ Reduced (S) such that $C' \subset C$ and $C' \neq C$. Also C is not a tautology.

Corollary 2. (Canonical Form Theorem). If S_1 and S_2 are any two sets of ground clauses which have identically the same models then Reduced $(\mathcal{R}^\infty(S_1))$ = Reduced $(\mathcal{R}^\infty(S_2))$.

Proof. Immediate from Corollary 1.

Note that this result says that if we have a set of interpretations $\mathcal{M}$ then for any set of ground clauses S, which has the interpretations

3. Intersection of a Set of Models

In this section we show that all literals in the intersection of a set of models $\mathcal{M}$ occur as unit clauses in $CF(\mathcal{M})$.

Example 2. If $\mathcal{M}$ consists of only one interpretation, say $\mathcal{M} = \{\{A,B,C,D\}\}$ then Reduced $(\mathcal{R}^{\infty}(S)) = \{[A],[B],[C],[D]\}$, for any set of clauses S which has that one interpretation as its only model. This shows that from any set of clauses S, which has only one interpretation, one may deduce by resolution unit clauses containing each literal in the model.

To see that Reduced $(\mathcal{R}^{\infty}(S)) = \{[A],[B],[C],[D]\}$ note that the set $S_1 = \{[A],[B],[C],[D]\}$ does in fact have only the one model. Also note that $\mathcal{R}^{\infty}(S_1) = S_1$ and hence Reduced $(\mathcal{R}^{\infty}(S_1)) = S_1$. By Corollary 2 this is unique and hence for any other set of clauses S if S has only the one model then Reduced $(\mathcal{R}^{\infty}(S))$ = Reduced $(\mathcal{R}^{\infty}(S_1)) = CF(\mathcal{M}) = \{[A],[B],[C],[D]\}$.

Next we would like to prove a theorem which has the above example as a special case.

Theorem 3. If $\mathcal{M}$ is a nonempty set of interpretations and $\Pi\mathcal{M}$ (i.e., the intersection of all interpretations in $\mathcal{M}$) is $\{L_1,\ldots,L_n\}$ and S is any set of clauses which has $\mathcal{M}$ as its only models, then one may deduce from S by resolution all of the unit clauses $[L_1],\ldots,[L_n]$.

Proof. Suppose $[L_1]$ is such a unit clause. Since $L_i \in \Pi\mathcal{M}$ it is in every interpretation in $\mathcal{M}$. Hence the unit clause $[L_i]$ is true in each interpretation in $\mathcal{M}$. But since $\mathcal{M}$ is the set of all models of S then it is clear that $S \Rightarrow [L_i]$. Hence by Theorem 1 there is

To see that the algorithm will produce $CF(\mathcal{M})$ note the following. In a way similar to the proof of Theorem 3 we can easily show that the set of clauses so produced does have $\mathcal{M}$ as its only models. Furthermore, any resolvent of a pair of clauses in the set will either be a tautology or some clause which we generated in the process. Thus it was put into the set when generated or was subsumed by some clause already produced. Thus, Reduced ($\mathcal{R}^{\infty}$(set of clauses produced by the algorithm from $\mathcal{M}$)) = set of clauses produced by the algorithm from $\mathcal{M}$ = $CF(\mathcal{M})$.

A number of examples of canonical forms for different sets of interpretation for 5 ground atoms follow in Appendix I as generated by our implementation of the above algorithm.

APPENDIX I

Examples of Canonical Forms for Sets of Models (Containing 5 Atoms)

MODEL SET

```
1 ((*T* A) (NIL B) (*T* C) (NIL D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((NIL B))
3 ((*T* C))
4 ((NIL D))
5 ((*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* D))
4 ((*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (NIL C) (NIL D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* E))
4 ((*T* C) (NIL D))
5 ((NIL C) (*T* D))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (*T* E))
3 ((*T* A) (*T* B) (*T* C) (NIL D) (NIL E))
4 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
5 ((*T* A) (*T* B) (NIL C) (NIL D) (*T* E))
6 ((*T* A) (NIL B) (*T* C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B) (*T* C))
3 ((*T* B) (*T* D))
4 ((*T* B) (*T* E))
5 ((*T* C) (*T* E))
6 ((NIL D) (*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (*T* E))
3 ((*T* A) (*T* B) (*T* C) (NIL D) (NIL E))
4 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
5 ((*T* A) (*T* B) (NIL C) (NIL D) (*T* E))
6 ((*T* A) (NIL B) (*T* C) (*T* D) (*T* E))
7 ((*T* A) (NIL B) (*T* C) (NIL D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B) (*T* C))
3 ((*T* B) (*T* E))
4 ((*T* C) (*T* E))
5 ((NIL D) (*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (*T* E))
3 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* E))
4 ((*T* C) (*T* D))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (NIL E))
3 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* C) (*T* D))
4 ((*T* C) (*T* E))
5 ((*T* D) (NIL E))
6 ((NIL D) (*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
3 ((*T* A) (*T* B) (NIL C) (NIL D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* E))
4 ((NIL C) (*T* D))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (NIL E))
3 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
4 ((*T* A) (NIL B) (*T* C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B) (*T* C))
3 ((*T* B) (*T* D))
4 ((*T* B) (*T* E))
5 ((*T* C) (*T* D))
6 ((*T* C) (*T* E))
7 ((*T* D) (NIL E))
8 ((NIL D) (*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (*T* E))
2 ((NIL A) (NIL B) (NIL C) (NIL D) (NIL E))
```

CANONICAL FORM

```
1 ((*T* A) (NIL B))
2 ((NIL A) (*T* B))
3 ((*T* A) (NIL C))
4 ((NIL A) (*T* C))
5 ((*T* A) (NIL D))
6 ((NIL A) (*T* D))
7 ((*T* A) (NIL E))
8 ((NIL A) (*T* E))
9 ((*T* B) (NIL C))
10 ((NIL B) (*T* C))
11 ((*T* B) (NIL D))
12 ((NIL B) (*T* D))
13 ((*T* B) (NIL E))
14 ((NIL B) (*T* E))
15 ((*T* C) (NIL D))
16 ((NIL C) (*T* D))
17 ((*T* C) (NIL E))
18 ((NIL C) (*T* E))
19 ((*T* D) (NIL E))
20 ((NIL D) (*T* E))
```

MODEL SET

```
1 ((*T* A) (*T* B) (*T* C) (*T* D) (NIL E))
2 ((*T* A) (*T* B) (*T* C) (NIL D) (*T* E))
3 ((*T* A) (*T* B) (NIL C) (*T* D) (*T* E))
```

CANONICAL FORM

```
1 ((*T* A))
2 ((*T* B))
3 ((*T* C) (*T* D))
4 ((*T* C) (*T* E))
5 ((*T* D) (*T* E))
6 ((NIL C) (NIL D) (NIL E))
```

APPENDIX II

UT-LISP Program Generating the Canonical Form of a Set of Models

```
DEFINE((


(CANFORM
 (LAMBDA(MS)
  (PROG (MI)
        (TERPRI)
        (TERPRI)
        (TERPRI)
        (TERPRI)
        (PRINT (QUOTE $$$ MODEL SET $))
        (TERPRI)
        (PRINL MS 0)
        (TERPRI)
        (TERPRI)
        (PRINT (QUOTE $$$ CANONICAL FORM $))
        (TERPRI)
        (COND
         ((SETQ MI (MISC MS))
          (SETQ MS (PULLUTS MI MS))
          (SETQ MI (WRAP MI))))
        (PRINL (GENCL MI (COMBI (REMATS (CAR MS)))) 0))))


(MISC
 (LAMBDA(MS)
  (MAPCON
   (CAR MS)
   (FQUOTE
    (LAMBDA(L)
     (SEARCH (CDR MS)
             (FQUOTE (LAMBDA (M) (NULL (MEMBER (CAR L) (CAR M))))
             NIL
             (FQUOTE (LAMBDA (Z) (LIST (CAR L))))))))))


(PULLUTS
 (LAMBDA(I MS)
  (MAPCAR
   MS
   (FQUOTE
    (LAMBDA(M)
     (PROG2 (MAP I (FQUOTE (LAMBDA (I) (SETQ M (REMOVE (CAR I) M)))))
            M))))))


(WRAP
 (LAMBDA (L) (MAPCAR L (FQUOTE (LAMBDA (L) (LIST L))))))


(REMATS
 (LAMBDA (M) (MAPCAR M (FQUOTE (LAMBDA (M) (CADR M))))))
```

```
(GENMODEL
 (LAMBDA(LL)
  (COND ((NULL LL) (LIST NIL))
        (T
         (APPEND
          (MAPCAR (SETQ ML (GENMODEL (CDR LL)))
                  (FQUOTE (LAMBDA (M) (CONS (LIST T (CAR LL)) M))))
          (MAPCAR
           ML
           (FQUOTE (LAMBDA (M) (CONS (LIST NIL (CAR LL)) M)))))))))


(SATMOD
 (LAMBDA(C MS)
  (SEARCH
   MS
   (FQUOTE
    (LAMBDA(MS)
     (SEARCH C
             (FQUOTE (LAMBDA (C) (MEMBER (CAR C) (CAR MS))))
             NIL
             (QUOTE TRUE))))
   NIL
   (QUOTE TRUE))))


(MODSET
 (LAMBDA(MS CS)
  (MAPCON
   MS
   (FQUOTE
    (LAMBDA(MS)
     (SEARCH
      CS
      (FQUOTE
       (LAMBDA(CS)
        (NULL
         (SEARCH (CAR CS)
                 (FQUOTE (LAMBDA (C) (MEMBER (CAR C) (CAR MS))))
                 (QUOTE TRUE)
                 NIL))))
      NIL
      (FQUOTE (LAMBDA (Z) (LIST (CAR MS)))))))))


(SUBSUMED
 (LAMBDA(C LC)
  (SEARCH LC
          (FQUOTE (LAMBDA (LC) (SUBSUMES (CAR LC) C)))
          (QUOTE TRUE)
          NIL)))


(SUBSUMES
 (LAMBDA(X Y)
  (SEARCH X
          (FQUOTE (LAMBDA (X) (NULL (MEMBER (CAR X) Y))))
          NIL
          (QUOTE TRUE))))
```

APPENDIX III

Working Documentation for a Program Generating the Canonical Form of a Set of Models

CANFORM (MS)

MS = set of models
Producing and printing out the Canonical Form of MS

MISC (MS)

MS = set of models
value = the intersection of all models in MS

PULLUTS (I MS)

I = the intersection of MS
MS = set of models
value = MS with I removed from each model

WRAP (L)

L = list
value = L with a new pair of parenthesis around each element

REMATS (M)

M = model
value = list of the atoms in M

COMBI (L)

L = list of n elements
value = list of all combinations of members of L from length 2 til length n

GENCL (RINF GL)

RINF = set of units whose literals belong to the intersection of a set of models MS
GL = all combinations of the remaining literals in the models of MS
value = the Canonical form of MS