PANEL '84
EXPODATA

DECIMA CONFERENCIA LATINOAMERICANA DE INFORMATICA

23 - 28 ABRIL '84
VIÑA DEL MAR - CHILE

• UNIVERSIDAD CATOLICA DE VALPARAISO
...NTRO LATINOAMERICANO DE ESTUDIOS EN INFORMATICA. CLEI

X CONFERENCIA LATINOAMERICANA de INFORMATICA

VIÑA DEL MAR   ABRIL 1984

documentos de trabajo

THEOREM PROVING IN CONSERVATIVE EXTENSIONS OF THEORIES

E. Lopez Passas, R. Lins de Carvalho
IME-PUC/RJ - BRASIL

ABSTRACT

This work includes some ideas and experiments about Automatic Theorem Proving. It introduce a new and complete strategy for Resolution based theorem provers for Definitional Theories, that is, the user can prove according to partial results. It is the strategy Translation by Level. [Passos,10].

1. INTRODUCTION

1.1 Basic Concepts

Some theories are "conservative extensions" of theories which have few proper axioms, by, for example, Boolean Algebras of Classes (BAC), where we have only two axioms [$\forall x\ (\neg x \in \emptyset)$] and [$\forall x \in V$)] and nine definitions. [Carvalho,16].

Definitions are statements which establish the expression meaning. A theory can be composed by its primitives and axioms about the primitivals, theorems would be valid relations among these primitives symbols. Thus, definitions are considered, as a point of view of Logics of first order, like a new axiom and, finally, it is considered that the introduction of new symbol serves only to facilitate the deduction of the structure of the theory and not to add them to that structure.

There are two basic criteria which may be observed by the following definitions:

Eliminability: A introducing a new symbol of a theory satisfies the criterion of eliminabilites if and only if: Whenever A1 is a formula in which the new symbol occurs, then there is a formula A2 in which the new symbol does not occur, such that $A \rightarrow (A1 \rightarrow A2)$ is derivable from the axioms and proceding definitions of the theory.

Non-creativity: A introducing a new symbol of a theory satisfies the criterion of non-creativity if and only if: there is no formula B in which the new symbol does not occur such that $(A \rightarrow B)$ is derivable from the axioms and proceding definitions of the theory, but B is not so derivable.

The rules for conditional definitions are on page 166 of [Suppes,14].

1.2 Purpose of the Work

The work has the purpose of studing certain theories or certain sets of formula which describe determinate situations or structures. For example, when it is needed describe situations or structures of Data Bank, kind of Data etc. in Computerization, many are not proper axioms, but definitions [Passos, Lanzelotte, Carvalho,6]. These definitions are important to facilitate the language. A complex formula, that is to say, one with so many symbols, can be reduced if definitions are introduced.

Some examples which will be mentioned belong to Mathematics because the structures are so much behaved and easy for describing what one wants. But anything impedes that when structures of Data Bank (see [6,17,18, 19,20], where there are several examples), Computers Networks or usual applications of Logics are described, there exist analogous situations, where the set of statements, at which its structures are described or specified, has some elements as Definitions, that is to say, this set of statements is really a Conservative Extension of Theory.

What the work fundamentally proposes is a system of automatic demostration of theorems, which uses a strategy called for us of Reduction for Translation by Level.

1.3 Example of the Aplication of the Strategy for Translations by Level, got From the Doctorate Thesis of [Passos,10] and [Carvalho,16].

Proof of Point Set Topology Theory

$$\forall X\ \forall Y\ \forall Z\ \forall W\ ((T(X, Y) \wedge Z \subset W) \rightarrow DER\ (X, Y, Z) \subseteq DER\ (X, Y, W)$$

The proof, by use of the strategy for translation by level.

1) T (X, Y)
2) Z C W
3) $\neg \overline{DER}$ (X, Y, W) $\subseteq$ DER (X, Y, W)
4) $\neg X \in Z \vee X \in W$ then (2)

It has been placed into prover that is found on [Passos,10], and with a given time and a limit to the generated clauses number it has not got to generate an empty clause □

Then it has been requested by the user the translation to next level of clause 3.

5) LIM (X, Y, Z, x) (1,3) translation of DER

It has not been got again therefore a translation to next level has been requested

6) $\neg$ NBH (X, Y, V, x) $\vee$ f1 (g(V), $\emptyset$) $\in V$
7) $\neg$ NBH (X, Y, V, x) C $\neg$f1 (g(V), $\emptyset$) = x translation of DER
8) $\neg$ NBH (X, Y, V, x) $\vee$ f1 (g(V), $\emptyset$) $\in$ z
9) $\neg$ x' $\in$ h4 (W, x), x) $\vee$ x' = x v x' $\in$ W
10) NBH (h4) (x, x), x)

where
x = f1 (DER (x, y, z), DER (x, y, w))
and $g(v) = (v - [x] \cap Z)$
11) NBH (h4 (W, x), x) from (6,7,8,9)
12) $\square$, empty clause, from (10,11)

2. CONSERVATIVE EXTENSIONS

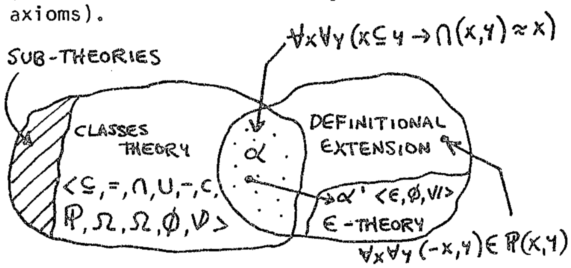
The classes elementary theory (Boolean Algebra of Classes) is an example of conservative extensions of a determined theory.

Example: Classes elementary theory)

A 1. $\forall x\ (\neg x \in \emptyset)$
A 2. $\forall x\ (x \in \mathcal{V})$
A 3. $\forall x\ \forall y\ (x \underline{C}\ y \leftrightarrow \forall z\ (z \in x \rightarrow z \in y))$
A 4. $\forall x\ \forall y\ (x \approx y) \leftrightarrow (X \underline{C}\ y \wedge y \underline{C}\ x))$
A 5. $\forall x\ \forall y\ \forall z\ (z \underline{\in \cap}\ (x,y) \leftrightarrow (x \in x \wedge z \in y))$
A 6. $\forall x\ \forall y\ \forall z\ (z \underline{\in \cap}\ (x,y) \leftrightarrow (z \in x \vee z \in y))$
A 7. $\forall x\ \forall y\ \forall z\ (z \underline{\in -}\ (x,y) \leftrightarrow (z \in x \wedge z \in y))$
A 8. $\forall x\ \forall y\ (x \in \overline{C}\ (y) \leftrightarrow x \in y)$
A 9. $\forall x\ \forall y\ (y \underline{\in} \overline{P}\ (x) \leftrightarrow y \underline{C}\ x)$
A10. $\forall x\ \forall y\ (y \underline{\in \mho}\ (x) \leftrightarrow \exists z\ (z \in x \wedge y \in z))$
A11. $\forall x\ \forall y\ (y \underline{\in \Omega}\ (x) \leftrightarrow \forall z\ (z \in x \rightarrow y \in z))$

In this theory only A1 and A2 are theory proper axioms and the others are definitions which are not necessaryly from functions, but of predicate. For example, intersection is not defined as a function, but as a predicate $\underline{\in \cap}$, by use of $\underline{\in \cap}$ as they were one symbol only.

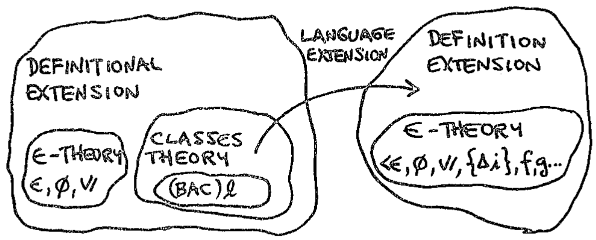
On figure below we see that in the Classes Theory there are sub-theories, and the intersection of the Classes Theory with a definitional extension of an $\in$ - theory, that one of proper axioms, would be a definitonal extension, with an α - formula not translated an a α' - formula translated by the strategy of level translation. α - formula should be written by use of definition elements only, and α', its translation in $\in$ - theory. Then this reduction is obtained from that intersection of Classes Theory with the extension of $\in$ - theory and a formula returns to $\in$ - theory, which formula only contains, as a unique predicate without symbols (except $\emptyset$, $\in$, and), one formula, which will be demostrated in $\in$ - theory (with only two axioms).



This can happen, in general, with an other sort object description and not in Set Theory necessaryly.

- theory is poor and not many interesting things one may say about it. But when translation is made, the definitional extension where there are so many valid formulas is used.

Classes Theory is a sub-set of definitional extension, as it showed on figure below.



3. ALGORITHM OF TRANSLATION BY LEVEL WITH ITS COMPLETUDE PROOF

3.1 Notation

Notation used in completive proving. R_0 is resolution with locking that we attribute for all literals the same indexis, therefore is equivalent to (resolution without locking)

$.S^A$ Axioms set of the theory

.ST Clauses obtained from the negation of the theorem

$.S^j = R_0^j\ (S^A\ US^T)$ for $j > 0$, $j \in \mathbb{N}$

$.S^0 = S_A^j\ U\ S = R_0^0\ (S^A\ U\ S^T)$ for $j = 0$

. SD definitions set

$.R_S^D$ definitional structure of the theory with S defintions

$.\ell$ maximum level of attempts

.N maximum level of R_S^D with respect to S

$.S_i^j = R^j\ (S_j^A\ U\ S^T\ U\ S_i)^D$ e $\ell > 1$ resolution with lock where i means that the clauses from S (S^D, S^T, ...) were indexed according to the scheme of indexing from considering the level N - i of $\mathbb{R}_S^D$.

3.2 Algorithm

Bellow we will describe the algorithm of refutation with resolution with translation by level, based on the definitional structure of theory.

The inputs for the algorithm are:

S^A, S^T, S^D, $\in$ N. We will represent the set of generated clauses by the algorithm for $\mathbb{R}_n\ (S^A\ , S^D\ , S^T\ , \ell)$.

Observations:

1. $j \leftarrow 0$; $K \leftarrow 0$; $S \leftarrow S^A\ U\ S^T : j_{max} \leftarrow \ell$; N (*1)
2. If $\square \in \mathbb{R}_k^j(S)$ then finish "T is a theorem" (*2)
If $\square \notin \mathbb{R}_k^j(S)$ and $R_k^j(S) = R_k^{j-1}(S)$ and $j > i$

then if $K = N + 1$,
finish "T ins't a theorem" (*3)
If not, go to (3) (*4)
(*5)
If $\square \in \mathbb{R}_k^j(S)$ and $\mathbb{R}_k^j(S)$ $R_k^j(S)$ if $j < j_{max}$
then do $j \leftarrow j+1$ and go to (2) (*6)
If not
If $K = N + 1$,
finish "T isn't a theorem" (*7)

3. Do $K \leftarrow K + 1$ $S = S'' \cup S_k^0$, $j \leftarrow 1$
and go to (2) (*8)

3.3 Observations about the steps of the algorithm

$R(S^A, S^T, S^D, \ell, N)$

(*1) Having in mind that even on a defitional theory some theorems (sentences to be tested) will have only primitive symbols, initially the set S on which we will be applied resolution, won't have any definition.

Definitions will only be introduced on the step, see observation, *8. K = 0 means not to use definitions. K grows in accord to the level related to the definitional structure.

(*2) Although the algorithm begins with j = 0, which means that we will test if $\square \in S$, that means a nonsense, some theories add as strategy the generation of inconsistency, soon after the process of defintions. For example: if we are on the "Theory of Sets" and we have a literal $(A \cup B) = \emptyset$ and a clause is $x \in A \cup B$, during the process of translation (introduction of definitions) we may introduce the clause $\square$ in the place of $x \in (A * B)$. This means to generate the empty clause before translating.

(*3) As K is equal to N + 1, there won't exist on the system, for j > 1, any definite symbol. And the fact of the resolution that will be unrestricted (equivalent not to use locking because all the indexes will be equal to 2, see "8) and hence the no increase of $\mathbb{R}_k^{j+1}(S)$ for resolution means that the system is satisfactory. This will always happen when the theory is decidable and, eventually in the case that the theory is undecidable. In both cases for a j sufficiently big.

(*4) Here although the resolution applied to the existent set of clauses is not permanent (no new clause was generated) we conclude that there are in the system defInte symbols, or the system is satisfactory. But now we can't known, unless we modify the algorithm to verify if there are definite symbols in $R_k^j(S)$. We can think that in the point of view of specification such aditional item is unnecessary (decidable theory is rare).

(*5) In this case the number of attempts of resolution is smaller than the user's permission, thus we must reapply resolution without introducing new definitions.

Note that if we use this algorithm interactively, the user, by the analysis of the clauses will be able to decide to modify j_{max}. Prohibiting resolution, going to (3) where there will be introduced new definitions.

(*6) A interactive system the user may modify ℓ, attributing to it a bigger value.

(*7) For $j_{max} = \ell$ given, it was not possible to verify, if T was a theorem. The user may change 1.

(*8) The introduction of new definition is done through a attribution of indexes to the literals $\mathbb{R}_k^j(S)$ getting the set S'. S_K^D by attribution of indexes to clauses of S_k according to the following scheme of attribution of index.

- S' is obtained from $R_k^j(S)$ by attributing index 2 to all of its literals which do not contain symbols of the level N - (K - 1) and index 1 to the other literals (which contain symbols of the level $N - (K_D - 1)$).

- S_K^D is obtained from S^D attributing to it index 1 to the first literal of clauses that define symbols of level, exactly, N - (K -1) and index 2 to the other literals of these clauses and index 3 to the literals of clauses of S^D that don't define symbols of level N - (K - 1).

Observation about the functioning of the algorithm.

Notice that the algorithm always ends, or by the exit of (*3) or by the exit (*2) that is the point of end that we are interested about, because we must show that we always go out through (*2).

And as we have already seen by the observation f (*5), any other exit is fictitious, because in the point of view of the theorem, it's equivalent to the hypotese "for some $\ell > 0$".

Notice too, that the minimum value of 1 for that, really, it may be possible to get, is at least equal to the minimum number of clauses used for the definition of the symbols in S.

In the proof of the theorem we limit ourselves to show that the algorithm always ends by the exit (2).

3.4 Theorem of Completive Rule

If $S^A \cup D^D \models T$ then $\square \in \mathbb{R}_n(S^A, S^D, S^T, \ell, N)$ for some ℓ, $\ell \in N$.

Demonstration

If $S^A \cup S^D \models T$ then

(As S^T are the clauses obtained by negation of T) $S^A \cup S^D S^T$ is inconsistent.

If in T don't occurrs symbols (N = 0) defined (S^D) then $S^A \models T$ and thus $S^A \cup S^T$ is incosistent thus $\square \in \mathbb{R}_n(S^A \cup S^T)$ so, to some j, $\square \in R^j$ $(S^A \cup S^T)$. As $\mathbb{R}_0 = \mathbb{R}$ (seen before), the algorithm stops with K = 0, on the step 2.

As lock resolution is refutationaly complete (Boyer's theorem), in [Chang,11], after K = N + 1 (all the definitions were used and so, the Symbols defined eliminated).

So, by the theory of definition

if $S^A \cup S^D \models T$, $S^A \models T'$, and by algorithm where T' is the obtained formula of T by the elimination of symbols defined by the

definitions in S^D, $S^A \cup S^{T'} \subseteq \mathbb{R}^j_{n+1} + 1(S)$.

For j sufficiently big, $S^A \cup S^{T'}$ is incosistent. And so for some $j' \geq j$,

$\square \notin \mathbb{R}^j_{n+1}(S)$.

4. CONCLUSION

What we have made up till now has been a theorem prover to some class of axiomatic theories that we call Definition Theories. New ideas included in the complete work are: generator of definitional theories and, Translation by level as a complete strategy for definitional Theories. The computerization system is interactive and, the experiments will be object of a next work.

5. BIBLIOGRAPHY

[1] Robinson, J.A. - Theorem proving on the computer - J. ACM 10, 1963.

[2] Robinson, J.A. - A machine-oriented logic based on the resolution principle - J. ACM, 12, 1965.

[3] Robinson, J.A. - A review of automatic theorem proving - Proceedings of Symposia in Applied Mathematics, vol. 19, 1967.

[4] Passos, E.P.L., de Carvalho, R.L. e Peixoto, S.R. - Communication Predicates: Complete Strategy for Resolution-Based Theorem-Provers an Evaluation of An Implementation - Current Topics in Cybernetics and Systems WOGSC, 1978, Springer-Verlag.

[5] Pion, M.M.B., de Carvalho, R.L. and Passos, E.P.L. - Interactive Systems to Construct Minimal Models on the Herbrand Universe, Publ. 1, 8, IME, 1980.

[6] Lanzelote, R.S.G., Passos, E.P.L. and de Carvalho, R.L. - MTD a Conversational System Oriented for Topology, EMCR - 80, Austria.

[7] Passos, E.P.L. and Silveira, G.C. - Quantifier Elimination to Mechanical Theorem Proving - Implementation - works from 3th Intern. Cong. of Cybernetics and System August 1975, Bucarest - Romenia, Publicado pela North-Holland em 1976.

[8] Passos, E.P.L. - Introduction to Mechanical Theorem Proving - Monographs in Computer Science and Computer Applications 2, 1971, INF-PUC/RJ.

[9] Passos, E.P.L. - Sistema Auxiliar do Matemático (S.A.M.) - Anais do 6° Painel de Discussão sobre Tópico de Computação, 1979, Valparaiso - Universidade Católica - Chile.

[10] Passos, E.P.L. - Algumas Idéias e Experimentos sobre Demonstração Automática de Teoremas - Tese de Doutorado, Sistemas - COPPE/UFRJ, 1981.

[11] Chang, C.L. and Lee, R.C.T. - Symbolic Logic and Mechanical Theorem Proving - New York, Academic Press, 1973.

[12] Margaris, A. - First Order Mathematical Logic - Waltham, Mass., 1967.

[13] Kreisel, G. and Krivine, J.L. - Elements of Mathematical Logic: Model Theory - Amsterdam, 1967.

[14] Suppes, P. - Introduction to logic - Prentice-Hall, New York, 1957.

[15] Wang, H. - Logic of Many Sorted Theories - JSL 17, 1952.

[16] de Carvalho, R.L. - "Some results in automatic Theorem proving", Ph.D. Thesis - Univ. Toronto, 1974.

[17] Minker, J. and Nicolas, J.M. - "On recursive axioms in dedutive database", Inf. Systems, vol 8, n° 1, 1983.

[18] Noble, H. - "The Automatic Generation of Test Data For A Relational Database", Inf. System, vol 8, n° 2, 1983.

[19] Castilho, J.M.V., Casanova, M.A. and Furtado, A.L. - "A Temporal framework for database specifications", Proceedings of Very Large Database Eight International Conference-Mexico, September 1982.

[20] Passos, E.P.L. - "Some results and experiments about Theorem proving", Proceeding of 5° Congress of Informatic and Automatic, Spain, May 1982.