SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 1, No. 3, November 2004, 15 - 20

Automated Reasoning-Alternative Methods

Aleksandar Perovic[1], Nedeljko Stefanovic[1], Milos Milosevic[1], Dejan Ilic[1]

Abstract: Our main goal is to describe a potential usage of the interpretation method (i.e. formal representation of one first order theory into another) together with quantifier elimination procedures developed in the GIS.

Keywords: Proving Methods.

1 Introduction

During the last few years, the Group for Intelligent Systems on the Faculty of Mathematics in Belgrade based its research on the quantifier elimination method. We are particularly interested in applications of this method in automated theorem proving. Some results are obtained (for example, see [2]), so we came to the natural question: Could we refine our method in some way, i.e. could we find the way to apply developed procedures to the broader class of theories.

We believe that one way to do this is to use the {\it interpretation method}, i.e. formal representation of one first order theory into another.

2 Definability

Let L be a first order language (i.e. L may contain some constant, function and relation symbols, but L can also be an empty set) and let T be a theory of language L (i.e. set of some sentences, where sentence is a first order formula which every variable is bounded with some quantifier).

To be *definable* in theory T means to be uniquely expressed (in T) by a first order formula of language L. Now we will give a strict definition for definability of constant, function and relation symbols:

- Each formula $\varphi(x)$ of language L such that T proves

$$\exists_1 x\varphi(x)$$

can be used for definition of new constant symbol c in the following way:

[1] Group for Intelligent Systems, Faculty of Mathematics, University of Belgrade

$$x = c \Leftrightarrow_{def} \varphi(x) .$$

- Each formula $\varphi(x_1,...,x_n, y)$ of language L such that T proves

$$\forall x_1 ... \forall x_n \exists_1 y \varphi(x_1,...,x_n, y)$$

can be used for definition of new n-ary function symbol F in the following way:

$$y = F(x_1,...,x_n) \Leftrightarrow_{def} \varphi(x_1,...,x_n, y) .$$

- Each formula $\varphi(x_1,...,x_n)$ of language L can be used for definition of new n-ary relation symbol R in the following way:

$$R(x_1,...,x_n) \Leftrightarrow_{def} \varphi(x_1,...,x_n) .$$

Note that each symbol of language L is definable in every theory T of the same language.

Example

Let $L = \{<\}$ be a language of linear orderings and let T be a theory of all sentences in language L that are true in the structure $(N,<)$, where $N = \{1,2,3,...\}$ is a set of all natural numbers and $<$ is usual ordering of N. Then, every natural number is definable in T. To see that, let us recursively define formulas $\varphi_0, \varphi_1,...$ as follows:

$$- \ \varphi_0(x) \Leftrightarrow_{def} \forall y (x \le y)$$

$$- \ \varphi_{n+1}(x) \Leftrightarrow_{def} \forall y (y \le x \Rightarrow \varphi_0(y) \vee ... \vee \varphi_n(y)) .$$

Note that formula $\varphi_0(x)$ asserts that x is minimum and formula $\varphi_{n+1}(x)$ asserts that x has exactly n predecessors. Thus, each natural number n is the only witness of formula $\varphi_{n-1}(x)$ in the structure $(N,<)$, so we have our claim.

As an interesting contrast to the previous example, let T^* be a theory of all sentences in language L of linear orderings which are true in the structure $(Z,<)$ (here Z is a set of all integers and $<$ is usual ordering of integers). It can be shown that none of the integers are definable in T^*.

To gain definability, it is sufficient to add one constant symbol to the language L.

3 Interpretation

Let L and L^* be first order languages, T be a theory of language L and T^* be a theory in language L^*. We say that language L is *interpretable* in a theory T^* if the following conditions hold:

- There is a definable in T^* unary predicate U such that T^* proves

$$\exists_1 x U(x) .$$

- For each constant symbol c of language L there is definable in T^* constant symbol c^* such that T^* proves

$$U(c^*) .$$

- For each n-ary function symbol F of language L there is definable in T^* function symbol F^* such that T^* proves

$$\forall x_1 ... \forall x_n (U(x_1) \wedge ... \wedge U(x_n) \Rightarrow U(F^*(x_1,...,x_n))) .$$

- For each n-ary relation symbol R of language L there is definable in T n-ary relation symbol R^* definable in T^*.

To obtain the well known interpretation theorem, first we are going to define a formula φ^* of language L^* for a given formula φ of language L by induction on the complexity of formula φ as follows:

- $(t_1 = t_2)^*$ is the formula $t_1 = t_2$, where t_1 are terms of language L;

- $(R(x_1,...,x_n))^*$ is the formula $(R^*(x_1,...,x_n))$, where R is n-ary relation symbol of language L;

- $(\neg\varphi)^*$ is the formula $\neg\varphi$;

- $(\varphi \wedge \psi)^*$ is the formula $\varphi^* \wedge \psi^*$;

- $(\exists x \varphi(x,...))^*$ is the formula $\exists x(U(x) \wedge \varphi^*(x,...))$.

We say that theory T is *interpretable* in T^* if there is an interpretation of language L into theory T^* such that for every nonlogical axiom φ in the theory T we have that T^* proves φ^*.

Theorem 1

Let L and L^* be a first order languages and let T and T^* be a theories in languages L and L^* respectively, such that T is interpretable in T^*. Then, for each formula φ of language L, if T proves φ then T^* proves φ^*.

The most persistent reader can find the proof of the staded theorem in [5].

The interpretation theorem gives us the essence of the method that we are developing.

So, for a given recursive theory T, suitable theory T^* should satisfy the following conditions:

- There is a recursive interpretation of axioms of T into T^* and there is a recursive interpretation of atomic formulas of T^* into T;

- T^* admits the quantifier elimination and we have an effective procedure for it;

- There is a recursive procedure for validity of quantifier free formulas in theory T^*.

So, we have the following algorithm:

input: formula φ of the language L

output: YES, if T proves φ; NO, otherwise

Step 1: Find an interpretation φ^* in language L* of a given formula φ.

Step 2: Find a quantifier free formula ψ of the Language L^* which is T^*-equivalent to formula φ.

Step 3: Check the validity for ψ. If ψ is valid in T^*, then output is YES, otherwise is NO.

Of course, the interpretation theorem guaranties the correctness of the algorithm stated above.

Example

Here we are briefly going to discuss an interpretation of monadic calculus in ZFC theory (ZFC states for Zermelo-Frankel set theory together with the axiom of choice). The embedding of the monadic calculus in the set theory is quite natural: for instance, syllogism Bocardo

Some M are not P

<u>Every M is S</u>

Some S are not P

can be expressed in ZFC as

$$M \setminus P \neq 0 \wedge M \subseteq S \Rightarrow S \setminus P \neq 0$$

(here we denoted empty set with 0). To be more precise, we will define a *monadic formula* in ZFC by induction on complexity as follows:

- Atomic formula is monadic formula;

- Boolean combination φ of monadic formulae is monadic formula if there are no variables x, y and z such that $x \in y$ and $y \in z$ are subformulas of φ;

- If $\varphi(x,...)$ is monadic formula and there is no variable y such that $y \in x$ is a subformula of φ, then $\exists x \varphi(x,...)$ and $\forall x \varphi(x,...)$ are also monadic formulas.

It is obvious that the set of all monadic formulas in ZFC is recursive. If we combine this with effective procedure of quantifier elimination for the monadic calculus developed in GIS (see [2]), we obtain a theorem prover for monadic formulas in ZFC.

4 Quantifier Elimination

Let T be a first order theory of language L. Recall that T admits the quantifier elimination if for every formula φ in language L exists quantifier free formula ψ of the same language T-equivalent to φ.

For the application of the method of quantifier elimination in decidability and automated theorem proving, see for example [2], [3] and [4].

Various teams of GIS researchers are developed programs for quantifier elimination in some first order theories (monadic calculus, algebraically closed fields etc.). We would like to appoint to the great importance of developing a program for quantifier elimination for the theory of real closed fields.

Such programs exists since 80-ies, but recent solutions are finding application in the robotics, control theory etc.

As an illustration of the fact that RCF (theory of real closed fields) admits the quantifier elimination, we will prove the fact that the following question is decidable:

Is given algebraic surface in the Euclidian space P^3 *rotational*?

Recall that algebraic surface in P^3 is given by the formula

$$p(x, y, z) = 0\ ,$$

where $p(x, y, z)$ is a polynomial over field of rational numbers Q.

Since every rational number is definable in language of fields of characteristic zero, we can conclude that $p(x, y, z) = 0$ is formula in the language of RCF. Now formula φ defined as

$$\exists a_1, a_2, a_3, b_1, b_2, b_3 \forall x_1, x_2, x_3 (p(x_1, x_2, x_3) = 0 \Rightarrow$$
$$\exists t \forall y_1, y_2, y_3 ((x_1 - a_1)^2 + (x_2 - a_2)^2 + (x_3 - a_3)^2 = (y_1 - a_1)^2 + (y_2 - a_2)^2 + (y_3 - a_3)^2 \wedge$$
$$\wedge\, a_1 (x_1 - y_1) + a_2\, (x_1 - y_1) + a_3\, (x_1 - y_1) = 0 \Rightarrow p(y_1, y_2, y_3) = 0))$$

asserts that algebraic surface S defined with

$$p(x, y, z) = 0$$

is rotational.

Since theory RCF admits quantifier elimination, there is quantifier free formula ψ in the language of RCF which is RCF-equivalent to φ.

Now it is sufficient to check validity of ψ in the P, which is decidable question for quantifier free formulas. So, given algebraic surface S in P^3 is rotational if and only if formula ψ holds in P.

5 Further Research

Recently we started to work on the project of developing software for proof assistant system, which will, beside standard procedures in automated reasoning (i.e. tableau and resolution), use quantifier elimination algorithms together with effective interpretations. We believe that some modules of this project will be available for download from the GIS web site (www.gisss.com) at the beginning of the winter semester.

6 References

[1] C. C. Chang, H. J. Keisler: Model Theory, Third Edition, North-Holland 1990.

[2] F. Marić, M. Marić, Ž. Mijajlović, A. Jovanović: Theorem provers based on the quantifier elimination method, Proc. XLVII ETRAN Conference, Herceg Novi, June 8-13, 2003, Vol. 3.

[3] Ž. Mijajlović: Decidable theories (in Serbian), Computer science, Number 1, Vol. 1, Beograd 1991.

[4] Ž. Mijajlović, A. Jovanović: Provers in algebra based on quantifier elimination, Proc. Of Algebraic Conference, Niš, 1996.

[5] J. Shoenfield: Mathematical Logic, Addison Wesley 1967.

[6] A. Jovanović, Ž. Mijajlović: Automated theorem proving in field theory using quantifier elimination, Filomat 1995.

[7] A. Jovanović, Ž. Mijajlović: Logic and databases, Proc. Of the Sixth International Symposium on Computer Science at the University of Dubrovnik, 1984.

[8] A. Jovanović: Group for Intelligent Systems – Problems and Results, Intelektualnie sistemy, Lomonossov Un. and RAN, 6, 2002.