ADMISSIBLE SUBSTITUTIONS IN SEQUENT CALCULI

A. V. Lyaletski

Abstract: *For first-order classical logic a new notion of admissible substitution is defined. This notion allows optimizing the procedure of the application of quantifier rules when logical inference search is made in sequent calculi. Our objective is to show that such a computer-oriented sequent technique may be created that does not require a preliminary skolemization of initial formulas and that is efficiently comparable with methods exploiting the skolemization. Some results on its soundness and completeness are given.*

Keywords: *completeness, first-order logic, quantifier rule, sequent calculus, skolemization, soundness*

Introduction

Investigations in computer-oriented reasoning gave rise to the appearance of various methods for the proof search in the classical 1st order logic. Particularly, sequent calculi were suggested by Gentzen [1]. But their practical application as a logical technique (without preliminary skolemization) of the intelligent systems has not received wide use: preference is usually given to the resolution-type methods. This is explained by higher efficiency of the resolution-type methods as compared to sequent calculi, which is mainly connected with different possible orders of the quantifier rule applications in sequent calculi while resolution-type methods, due to skolemization, are free from this deficiency.

In its turn, the deduction process in sequent calculi reflects sufficiently well natural theorem-proving methods which, as a rule, do not include preliminary formula skolemization so that reasonings are performed within the scope of the signature of the initial theory. This feature of sequent calculi becomes important when some interactive mode of proof is developed since it is preferable to present the output information concerning the proof search in the form usual for man. That is now the problem of the efficient quantifier manipulation makes its appearance.

When quantifier rules are applied, some substitution of selected terms for variables is made. To do this step of deduction sound, certain restrictions are put on the substitution. The substitution, satisfying these restrictions, is said to be admissible. Here we investigate the classical notion of admissible substitution and show how it can be modified so that efficient sequent calculi can be finally obtained. We use the calculus G [2] for the demonstration of the way of the construction of such a modification denoted by mG here. Note that when constructing mG, we don't touch upon any procedure of selection of propositional rules and terms substituted, focussing our attention on quantifier handling only.

Genzen’s Notion of Admissible Substitutions

Classical quantifier rules, substituting arbitrary structure terms when applied “from bottom to top", are usually of the following form [2]:

$$\frac{\Gamma_1, A[t/x],\ \forall xA, \Gamma_2 \rightarrow \Gamma_3}{\Gamma_1, \forall xA, \Gamma_2 \rightarrow \Gamma_3} \quad (\forall\text{: left})$$

$$\frac{\Gamma_1 \rightarrow \Gamma_2, A[t/x], \exists xA, \Gamma_3}{\Gamma_1 \rightarrow \Gamma_2, \exists xA, \Gamma_3} \quad (\exists\text{: right})$$

where the term t is required to be free for the variable x in the formula A. This restriction of the substitution of t for x gives Gentzen’s (classical) notion of an admissible substitution, which proves to be sufficient for the needs of the proof theory. But it becomes useless from the point of view of efficiency of computer-oriented theorem-proving methods. It is clear from the following example.

Consider a sequent $A_1, A_2 \rightarrow B$, where A_1 is $\forall x_1 \exists y_1(R_1(x_1) \vee R_2(y_1))$, A_2 is $\forall x_2 \exists y_2(R_1(y_2) \vee R_2(x_2))$, and B is $\exists x_3 \forall y_3(R_2(x_3) \vee R_3(y_3))$. The provability of this sequent in calculus G will is established below, while here we notice that quantifier rules must be applied to all the quantifiers occurring in A_1, A_2, and B. Therefore, classical notion of admissible substitution yields 90 (= 6!/(2!*2!*2!)) different orders of the quantifier rule applications ("from bottom to top") to the sequent $A_1, A_2 \rightarrow B$. It is clear that resolution type methods allow avoiding this redundant work.

Kanger's Notion of the Admissible Substitutions

To optimize procedure of the applications of quantifier rules, S.Kanger suggested in [2] his calculus of Gentzen type, denoted here by K. In calculus K a "pattern" of a deduction tree is first constructed with the help of special variables, the so called parameters and dummies. At some times an attempt is made to convert a "pattern" into proof tree to complete the deduction process. In case of failure, the process is continued.
The main difference between K and G consists in a special modification of the above quantifier rules and in a certain splitting (in K) of the process of the "pattern" construction into stages. In K the rules ($\forall$: left) and ($\exists$: right) are of the following form:

$$\frac{\Gamma_1, A[d/x], \forall xA, \Gamma_2 \rightarrow \Gamma_3}{\Gamma_1, \forall xA, \Gamma_2 \rightarrow \Gamma_3} \qquad d/t_1,...,t_n$$

$$\frac{\Gamma_1 \rightarrow \Gamma_2, A[d/x], \exists xA, \Gamma_3}{\Gamma_1 \rightarrow \Gamma_2, \exists xA, \Gamma_3} \qquad d/t_1,...,t_n$$

where $t_1,...,t_n$ are the terms occurring in the conclusion of the rules, d is the dummy, and $d/t_1,...,t_n$ denotes that when an attempt is made to convert "pattern" into proof tree, the dummy d must be replaced by one of the terms $t_1,...,t_n$. The replacement of dummies by terms is made in the end of every stage, and at every stage the rules are applied in a certain order.
This scheme of the deduction construction in calculus K leads to a notion of the Kanger-admissible substitution, which is more efficient than the classical one. Thus in the above example it yields only 6 (=3!) variants of different possible orders of the quantifier rule applications (but none of these variants is preferable). Despite this, the Kanger-admissible substitutions still did not allow to attain the efficiency comparable with that when the skolemization is made. It is due to the fact that, as in case of the classical admissible substitution, it is required to select a certain order of the quantifier rule applications when an input sequent is deduced, and, if it proves to be unsuccessful, the other order of applications is tried, and so on.

New Notion of Admissible Substitutions

For constructing the modification mG of calculus G from [2], let us introduce a new notion of admissible substitutions in order to get rid of the dependence of the deduction efficiency in sequent calculi on different possible orders of quantifier rule applications. The main idea is to determine, proceeding from quantifier structures of formulas of an input sequent and a substitution under consideration, would there exists a sequence of desired quantifier rule application. (This notion was used in slightly modified form in [3].)
Substitution is defined as a finite (maybe, empty) set of ordered pairs, every of which contains a variable and a term and is written in the form t/x, where x is the variable and t is the term of substitution [4].
We assume that besides usual variables there are two countable sets of special variables, namely of parameters and dummies.
Let P be a set of sequences of parameters and dummies, and s be a substitution. Put T(P,s) = {<z,t,p>: z is the variable of s, t is the term of s,p P, and z lies in p to the left of some parameter from t}. The substitution s is said to be admissible for P if and only if (1) the variables of s are only dummies and (2) in T(P,s) there are no elements $<z_1,t_1,p_1>$,..., $<z_n,t_n,p_n>$ such that $t_2/z_1 \in s,...,t_n/z_{(n-1)} \in s$, $t_1/z_n \in s$ (n>0).

Calculus mG

As in the case of calculus G, its modification mG deals with formulas, except that in mG every formula from a sequent has a certain sequence of parameters and dummies. Therefore, it is convenient to define calculus mG by means of the pairs <p,A>, where A is the formula and p - the sequence (word) of parameters and dummies. Also, it will be assumed that the empty sequence is always added to all formulas from the input sequent (that is, from the sequent to be proved).
The rules of the calculus mG are the following.

Propositional rules:

$$\frac{\Gamma_1, <p,A>, <p,B>, \Gamma_2 \rightarrow \Gamma_3}{\Gamma_1, <p, A \wedge B>, \Gamma_2 \rightarrow \Gamma_3} \qquad \frac{\Gamma_1 \rightarrow \Gamma_2, <p,A>, \Gamma_3 \quad \Gamma_1 \rightarrow \Gamma_2, <p,B>, \Gamma_3}{\Gamma_1 \rightarrow \Gamma_2, <p, A \wedge B>, \Gamma_3}$$

$$\frac{\Gamma_1,\langle p,A\rangle,\Gamma_2\to\Gamma_3 \quad \Gamma_1\to\Gamma_2,\langle p,B\rangle,\Gamma_3}{\Gamma_1,\langle p,A\vee B\rangle,\Gamma_2\to\Gamma_3} \qquad \frac{\Gamma_1\to\Gamma_3,\langle p,A\rangle,\langle p,B\rangle,\Gamma_2}{\Gamma_1\to\Gamma_2,\langle p,A\vee B\rangle,\Gamma_3}$$

$$\frac{\Gamma_1,\Gamma_2\to\langle p,A\rangle \quad \langle p,B\rangle,\Gamma_1,\Gamma_2\to\Gamma_3}{\Gamma_1,\langle p,A\supset B\rangle,\Gamma_2\to\Gamma_3} \qquad \frac{\langle p,A\rangle,\Gamma_1\to\Gamma_3,\langle p,B\rangle,\Gamma_2}{\Gamma_1\to\Gamma_2,\langle p,A\supset B\rangle,\Gamma_3}$$

$$\frac{\Gamma_1,\Gamma_2\to\langle p,A\rangle,\Gamma_3}{\Gamma_1,\langle p,\neg A\rangle,\Gamma_2\to\Gamma_3} \qquad \frac{\langle p,A\rangle,\Gamma_1\to\Gamma_2,\Gamma_3}{\Gamma_1\to\Gamma_2,\langle p,\neg A\rangle,\Gamma_3}$$

Quantifier rules:

$$\frac{\Gamma_1,\langle pd,A[d/x],\rangle,\langle p,\forall xA\rangle,\Gamma_2\to\Gamma_3}{\Gamma_1,\langle p,\forall xA\rangle,\Gamma_2\to\Gamma_3} \qquad (\forall\text{: left'})$$

$$\frac{\Gamma_1,\to\Gamma_2,\langle pd,A[d/x]\rangle,\langle p,\exists xA\rangle,\Gamma_3}{\Gamma_1,\to\Gamma_2,\langle p,\exists xA\rangle,\Gamma_3} \qquad (\exists\text{: right'})$$

$$\frac{\Gamma_1\to\Gamma_2,\langle pz,A[z/x]\rangle,\Gamma_3}{\Gamma_1\to\Gamma_2,\langle p,\forall xA\rangle,\Gamma_3} \qquad (\forall\text{: right'})$$

$$\frac{\Gamma_1,\langle pz,A[z/x]\rangle,\Gamma_2\to\Gamma_3}{\Gamma_1,\langle p,\exists xA\rangle,\Gamma_2\to\Gamma_3} \qquad (\exists\text{: left'})$$

Here d is a new dummy, z is a new parameter, p is a sequence of parameters and dummies, Γ_1, Γ_2, and Γ_3 are arbitrary sequences of pairs, consisting of sequences (of dummies and parameters) and formulas, A, B are arbitrary formulas.

Applying first rules "from bottom to top" to the input sequent and afterwards to its "heirs", and so on, we finally obtain a so-called deduction tree.

A deduction tree D is called a proof tree for the input sequent (in mG) if and only if there exists a substitution of terms for variables, s, such that (1) s is admissible for set of all sequences of parameters and dummies from D and (2) after application of s to the formulas from all upper sequents of D we obtain axioms, that is, the sequents $\Gamma_1\to\Gamma_2$ such that Γ_1 and Γ_2 contain a common formula.

The main result concerning the calculus mG is as follows.

Theorem. Let $A_1,\ldots,A_m,B_1,\ldots,B_n$ be the formulas of the 1st order language. There exists a proof tree for the input sequent $\langle ,A_1\rangle,\ldots,\langle ,A_m\rangle\to\langle ,B_1\rangle,\ldots,\langle ,B_n\rangle$ in calculus mG if and only if there exists a proof tree for the input sequent $A_1,\ldots,A_m\to B_1,\ldots,B_n$ in calculus G.

Proof.

(=>) Let D be a proof tree for the input sequent $\langle ,A_1\rangle,\ldots,\langle ,A_m\rangle\to\langle ,B_1\rangle,\ldots,\langle ,B_n\rangle$ in the calculus mG, and s be a substitution, which converts all upper sequents of D into axioms and is admissible for set P of all sequences of parameters and dummies from D. Without any loss of generality, we may assume that terms of s do not contain dummies for otherwise these dummies could be replaced by a constant, say, c0.

Since s is admissible for P, it is possible to construct the following sequence p consisting of parameters and dummies which form the sequences of P:

(i) every p' P is a subsequence of p, and

(ii) the substitution s is admissible for {p} (i.e. there is no an element $\langle z,t,p\rangle$ T({p},s) such that $t/z\in s$.

Such a sequence p may be generated, for example, by the convolution algorithm from [3], applied to a list of all the sequences from P provided that in the convolution algorithm are treated parameters as existence quantifiers, and dummies universal quantifiers.

Property (i) of the sequence p and formulation of the propositional and quantifier rules permit to make the following assumption:

When D was constructed, propopositional and quantifier rules were applied ("from bottom to top") in the order that corresponds to looking through p from the left to right: i.e. when the first quantifier rule was applied, the first variable (a parameter or a dummy) of p was generated, when the second quantifier rule was applied, the second variable of p was generated, and so on.

Now it is possible to covert the tree D into proof tree D' for the input sequent $A_1,...,A_m \rightarrow B_1,...,B_n$ in calculus G. To do this, let us "repeat" the process of the construction of D in the above order p and execute the following transformations:

1) Suppose that in a processed node of D one of the following rules was applied:

Γ_1, <pd,A[d/x],>,<p,$\forall$xA>,$\Gamma_2 \rightarrow \Gamma_3$ ($\forall$: left')

Γ_1,<p,$\forall$xA>,$\Gamma_2 \rightarrow \Gamma_3$

or

Γ_1, $\rightarrow \Gamma_2$,<pd,A[d/x]>,<p,$\exists$xA>, Γ_3 ($\exists$: right')

Γ_1, $\rightarrow \Gamma_2$,<p,$\exists$xA>, Γ_3

and t/d s for some term t. The term t is free for d in A, because the order of applications of quantifier rules is reflected by p, and property (ii) is satisfied. Therefore, the admissibility in the classical sense will be observed when the above rules ($\forall$: left') and ($\exists$: right') are replaced in D by rules ($\forall$: left) and ($\exists$: right) of the calculus G: and all other occurrences of d in D are replaced by t.

Γ_1, A[t/x], $\forall$xA, $\Gamma_2 \rightarrow \Gamma_3$ ($\forall$: left)

Γ_1,$\forall$xA, $\Gamma_2 \rightarrow \Gamma_3$

or

$\Gamma_1 \rightarrow \Gamma_2$, A[t/x], $\exists$xA, Γ_3 ($\exists$: right)

$\Gamma_1 \rightarrow \Gamma_2$, $\exists$xA,Γ_3

2) In other cases the rules of the calculus mG are replaced by their analogs from G by a simple deleting of sequences of parameters and dummies from these rules.

It is evident that D' is a deduction tree in the calculus G. Furthermore, the way of conversion of D into D' allows making the conclusion that upper sequents of D' are axioms of the calculus G. Thus, D' is a proof tree for the input sequent $A_1,...,A_m \rightarrow B_1,...,B_n$ in G.

(<=) Let D' be a proof tree for the input sequent $A_1,...,A_m \rightarrow B_1,...,B_n$ in G. Convert D' into tree D, which, as be can seen bellow, is a proof tree for the input sequent <,A_1>,...,<,A_m> $\rightarrow$ <,B_1>,...,<,B_n> in mG. For this purpose "repeat" ("from bottom to top") a process of construction of D', replacing in D' every rule application by its analog in mG and subsequently generating substitution s. (Initially s is the empty substitution.)

1) If an applied rule is one of the following:

Γ_1, A[t/x], $\forall$xA, $\Gamma_2 \rightarrow \Gamma_3$ ($\forall$: left)

Γ_1,$\forall$xA, $\Gamma_2 \rightarrow \Gamma_3$

or

$\Gamma_1 \rightarrow \Gamma_2$, A[t/x], $\exists$xA, Γ_3 ($\exists$: right)

$\Gamma_1 \rightarrow \Gamma_2$, $\exists$xA,Γ_3

then it is replaced by

Γ_1, <pd,A[d/x],>,<p,$\forall$xA>,$\Gamma_2 \rightarrow \Gamma_3$ ($\forall$: left')

Γ_1,<p,$\forall$xA>,$\Gamma_2 \rightarrow \Gamma_3$

or

Γ_1, $\rightarrow \Gamma_2$,<pd,A[d/x]>,<p,$\exists$xA>, Γ_3 ($\exists$: right')

Γ_1, $\rightarrow \Gamma_2$,<p,$\exists$xA>, Γ_3

accordingly with adding t/d to the existing substitution s, where d is a new dummy, and with substituting d for those occurrences of t into "heirs" of the formula A[t/x], which appeared as a result of applying of a replaced rule "inserting" the term t.

2) In all other cases replacement of the rules of G by the rules of mG is evident. (Note that $<,A_1>,...,<,A_m> \rightarrow <,B_1>,...,<,B_n>$ is declared as input sequent of D. The rules ($\exists$: left) and ($\forall$: right) may be considered as those inserting new parameters).

Since D' is a proof tree in the calculus utilizing the classical notion of admissible substitution, then it is clear that the finally generated substitution s is admissible (in the new sense) for a set of all sequences of parameters and dummies from D. Therefore, D is a proof tree for the input sequent $<,A_1>,...,<,A_m> \rightarrow <,B_1>,...,<,B_n>$ in mG. *Q.E.D.*

Corollary 1. For any formulas $A_1 ,..., A_m, B_1 ,..., B_n$ the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $<,A_1>,...,<,A_m> \rightarrow <,B_1>,...,<,B_n>$ in calculus mG.

Proof.

In accordance with [2] the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $A_1,...,A_m \rightarrow B_1,...,B_n$ in the calculus G. On the basis of the Theorem the latter condition holds true if and only if a proof tree for the input sequent $<,A_1>,...,<,A_m> \rightarrow <,B_1>,...,<,B_n>$ can be constructed in calculus mG. *Q.E.D.*

To demonstrate the deduction technique, consider the sequent $A_1,A_2 \rightarrow B$ from the above example and establish its provability in calculus G. To do this, construct a proof tree for the input sequent $<,A_1>,<,A_2> \rightarrow <,B>$ in calculus mG and use the Theorem.

Applying to the initial sequent only quantifier rules we can receive the following sequent:

$<d_1z_1,R_1(d_1) \vee R2(z_1)>,<,A_1>,<d_2z_2,R_1(z_2) \vee R_2(d_2)>,<,A_2> \rightarrow <d_3z_3,R_2(d_3) \vee R_3(x_3)>, <d_3z_3, R_2(d_3) \vee R_3(x_3)>,<,B>$, where $d_1,...,d_4$ are dummies, $z_1,...,z_4$ are parameters.

Now let us apply prorositional rules to the last sequent as long as they are applicable. As a result, we get a deduction tree D. If we generate the substitution $s = \{z_2/d_1, z_3/d_2, c_0/d_3, z_1/d_4\}$ (c_0 is a constant), then we can draw the following conclusions concerning s and D:

1) s is admissible for the set of all sequences of dummies and parameters from D, and

2) every upper sequent from D may be transformed into axioms by applying of s to it.

So, in accordance with the above Theorem the sequent $A_1,A_2 \rightarrow B$ is provable in the calculus G. *Q.E.D.*

Some Reconstruction of mG

The formulation of the calculus mG shows that the order of the quantifier rule applications is immaterial. In the calculus mG the quantifier rules are needed to determine a quantifier structure of formulas from the input sequent. This observation gives us possibility to construct a modification mG' of the calculus mG, which contains the so-called doubling rules instead of all the quantifier rules.

Doubling rules:

$\Gamma_1,<pdz_1...z_k,A>,<pd'u_1...u_k,A[d'/d,u_1/z_1,...,u_k/z_k]>, \Gamma_2 \rightarrow \Gamma_3$ (D: left)

$\Gamma_1,<pdz_1...z_k,A>,\Gamma_2 \rightarrow \Gamma_3$

$\Gamma_1 \rightarrow \Gamma_2,<pdz_1...z_k,A>,<pd'u_1...u_k,A[d'/d,u_1/z_1,...,u_k/z_k]>,\Gamma_3$ (D: right)

$\Gamma_1 \rightarrow \Gamma_2,<pdz_1...z_k,A>,\Gamma_3$

Here p is a sequence (maybe, empty) of parameters and dummies, the most right variable of which (in non-empty case) is a parameter, d is a dummy, for i=1,...,k z_i is a dummy or parameter, and u_i is a new dummy or a parameter (in accordance with z_i).

In calculus mG' a deduction process starts with an input sequent of the form: $<p_1,M_1>,..., <p_m,M_m> \rightarrow <q_1,N_1>,...,<q_n,N_n>$, where $M_1 ,..., M_m, N_1 ,..., N_n$ are formulas without quantifiers, and $p_1,...,p_m,q_1,...,q_n$ are sequences of parameters and dummies, which are determined by the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$, tested for validability, by the following way:

Let $A'_1 , ... , A'_m, B'_1, ..., B'_n$ be some prefix normal forms of the formulas $A_1, ..., A_m, B_1, ..., B_n$, respectively. Then for every i=1,...,m (j=1,...,n) M_i is a matrix of A'_i (N_j is a matrix of B'_j), and p_i (q_j) is obtained by means of replacing in prefix of A'_i (B'_j) of every universal (existential) quantifier by a new dummy and of every existential (universal) quantifier by a new parameter.

All other notions (addmissible substitutions, deduction trees, proof trees, and so on) are the same as in the case of the calculus mG.

Corollary 2. For any formulas A_1, ..., A_m, B_1, ..., B_n the formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if there exists a proof tree for the input sequent $<p_1,M_1>,...,<p_m,M_m> \rightarrow <q_1,N_1>,...,<q_n,N_n>$ in the calculus mG'.

Proof.

The formula $(A_1 \wedge ... \wedge A_m) \supset (B_1 \vee ... \vee B_n)$ is valid if and only if $(A'_1 \wedge ... \wedge A'_m) \supset (B'_1 \vee ... \vee B'_n)$ is valid, where A'_1 , ... , A'_m, B'_1, ..., B'_n are prefix normal forms of A_1, ..., A_m, B_1, ..., B_n, respectively. It is easy to see that a proof tree for the input sequent $<p_1,M_1>,...,<p_m,M_m> \rightarrow <q_1,N_1>,...,<q_n,N_n>$ in mG' may be constructed on the basis of a proof tree for the input sequent $<,A'_1>, ... , <,A'_m> \rightarrow <,B'_1>,...,<,B'_n>$ and vice versa. To complete the proof, use Corollary 1. *Q.E.D*.

Remark. In calculus mG', the quantifier structures of formulas A_1, ..., A_m, B_1, ..., B_n are taken into account by means of sequences p_1, ..., p_m, q_1, ..., q_n. Selection of sequences for determination of quantifier dependencies does not play a principal role and was made for the purpose of visualizing and simplifying of the subject matter. It is possible to construct a (correct and complete) version of calculus mG' using analogs of "schemes" [5] instead of sequences (which also consist of parameters and dummies and reflect the quantifier structures of initial formulas more exactly) and modifying the rules (D: left) and (D: right). Observe also that Herbrand theorem in the form A from [5] may be easily obtained on the basis of a correctness and completeness of the version of calculus mG'.

Conclusion

In this paper the questions of implementation of computer-oriented sequent calculi are not considered because the development of efficient calculi requires optimizing the order of the propositional rule applications and selecting a method for generating of terms which may produce a proof tree. Bypassing details observe that for this purpose the unification algorithm combined with the introduced notion of admissible substitution is suitable. It was the approach that investigated at the level of modern vision [7] of the Evidence Algorithm programme, EA, advances by V. Glushkov. By now, the first version of the System for Automated Deduction, SAD, has been implemented (see Web-site 'http://ea.unicyb.kiev.ua'). This implementation is based on a number of papers devoted to EA and SAD (see, for example, [8-10]).

Bibliography

1. Gentzen G. Untersuchungen uber das Logische Schliessen. Math. Zeit., 39, 1934, 176-210.
2. Gallier J.H. Logic for Computer Science: Foundations of Automatic Theorem Proving. - New York: Harper and Row, Inc., 1986, 513 pp.
3. Kanger S. Simplified Proof Method for Elementary Logic. - Comp. Program. and Form. Sys.: Stud. in Lodic, Amsterdam: North-Holland, Publ. Co., 1963, p. 87-93.
4. Lyaletski A.V. Variant of Herbrand Theorem for Formulas in Prefix Form (in Russian). – Kibernetika, 1981, No 1, p. 112-116.
5. Robinson J. A Machine-Oriented Logic Based on Resolution Principle. - J. of the ACM, 1965, p. 23-41.
6. Herbrand J. Recherches sur la Theorie de la Demonstration. - Travaux de la Societe des Sciences et de Lettres de Varsovie, Class III, Sciences Mathematiques et Physiques, 1930, 33.
7. Kapitonova Y., Letichevsky A., Lyaletski A., and Morokhovets. Algoritm Ochevidnosti - 2000 (project). - Proc. of the 1st Int. Conf. UkrPROG'98, Kiev, Ukraine, 1998, 68-70.
8. Degtyarev A., Lyaletski A., and Morokhovets M. Evidence Algorithm and Sequent Logical Inference Search. - Lecture Notes in Artificial Intelligence, 1705, 1999, 99-117.
9. Degtyarev A., Lyaletski A., and Morokhovets M. On the EA-Style Integrated Processing of Self-Contained Mathematical Texts.-Symbolic Computation and Automated Reasoning (the book devoted to the CALCULEMUS-2000 Symposium: edited by M. Kerber and M. Kohlhase), A K Peters, Ltd, USA, 2001, 126-141.
10. Verchinine K., Degtyarev A., Lyaletsky A., and Paskevich A. System for Automated Deduction (SAD): Linguistic and Deductive Peculiarities. - Advances in Soft Computing (Intelligent Information Systems 2002. (M.A.Klopotek, S.T.Wierzchon, M.Michalewicz (eds)). Physica/Springer Verlag, Heidelberg New York, ISBN 3-7908-1509-8, 2002, 413-422.

Author information

Alexander V. Lyaletski – Faculty of Cybernetics, Kiev Nashional Taras Shevchenko University, Senior Scientist Researcher; 2, Glushkov avenue, building 6, 03022 Kiev, Ukraine; e-mail: lav@unicyb.kiev.ua