Mathematical Text Processing in EA-style: a Sequent Approach

ALEXANDER LYALETSKI
Senior researcher of the Faculty of Cybernetics, Taras Shevchenko National University of Kyiv, Kyiv, Ukraine

The paper devoted to the logical aspect of mathematical text processing satisfying such principles of the so-called Evidence Algorithm as that the syntactical form of a task under consideration should be preserved and proof search should be proceeded in the signature of an initial theory, that is it should be carried out without performing preliminary skolemization being a forbidden operation for a number of logics. It contains an approach to construct computer-oriented cut-free sequent calculi for classical and intuitionistic first-order logics as well as their modal extensions (without or with equality). The approach exploits the original notions of admissibility and compatibility, which allows avoiding the dependence of proof search in sequent calculi on different orders of quantifier rule applications. Following it, quantifier-rule-free sequent calculi are constructed. Results on the coextensivity of these calculi with Kanger-type sequent calculi are given.

> "... the computer can significantly extend human capabilities in the area of establishing new mathematical (and not only mathematical) facts. Doing a little dreaming, one can talk about the times when fruitful creative work in mathematics and other exact sciences will be impossible without the usage of computers, and the success of a scientific study will be determined primarily by the craft of programming a strategy for scientific research..."[1]
>
> V.M. Glushkov (1957)

1. INTRODUCTION

In Kyiv, investigations in automated reasoning were initiated by V.M. Glushkov at the beginning of 1960s, when the first team on formalized mathematical text processing was gathered. Being a mathematician and algebraist, from the very outset of his activity in Cybernetics, he first of all was interesting in the problem of doing mathematics with the help of automated theorem proving in formal theories.

V.M. Glushkov formulated this problem in a slightly unusual way. Let us consider a relatively well formalized mathematical theory, e.g. the group theory. There are a small number of basic facts (axioms) which are considered to be evident even for beginners. After applying simple purely logical tools, one obtains several consequences. They are also evident. Then one can apply the same logical tools to the conclusions and so on. Are the results still evident? If the conclusions were obtained by a programmed inference engine, the answer is "yes, they are". From the viewpoint of this engine. However, probably not from the human point of view. Thus, provided the above-mentioned engine, one is able to prove/verify something that is not evident for humans. Further to that, this "evidence-maintaining engine" may be reinforced with heuristics, proof methods, lemma application, definition expansion, and so on. In this way, the notion of "being evident" could be enlarged to the extent that might include nontrivial facts/theorems.

This approach to automated theorem proving found its reflection in the so-called Evidence Algorithm programme, EA, announced by V.M. Glushkov in [2], where he wrote:

"... a good (and maybe, the best) problem of the simulation [of human activity] might be the problem of automated theorem proving in mathematics. The center of the study in this field should be displaced from constructing "universal" theorem-proving programs towards creating programming and operating systems that (if necessary) allow rapid programming of proof search even for a unique difficult theorem and that, in the case of necessity, are capable to interact with a mathematician for proving this theorem in real time."

This Glushkov's point of view on automated theorem proving has lead to making simultaneous investigations on formalized mathematical texts processing, including the creation of formalized languages for their presentation in the form most appropriate for a user, formalization and evolutional (evidential) development of a computer-made proof step, construction of an EA information environment having an influence on a current evidence of a computer-made proof step, and interactive man-assistant search of a proof. So, we see that Glushkov's approach shares certain ideas and has some common features with the QED manifesto [3] as well as with Mizar [4] and Calculemus [5] projects.

2. EA APPROACH TO MATHEMATICAL TEXT PROCESSING

A lot of investigations in automated theorem proving have being made since the appearance of the Evidence Algorithm, EA, the modern vision of which was called the evidential paradigm in [6]. As a result, there were constructed the Russian-language [7] (1978) and English-language [8] (2002) SAD systems[1] reflecting the EA requirements to formalized mathematical text processing. Both the systems have similar structures and are based on the above-given general EA principles. (At that, accordingly to EA, the main efforts of the EA developers were primarily directed towards the development, investigation, and implementation of linguistic tools and logical methods of proof search in first-order logics.)

The Russian and English SAD systems operate according to the following chains of formalized mathematical text transformations (a more detailed description of the Russian and English SADs can be found in [9]):

Text prepared by a user for proving/verifying a theorem ===========>
(using a parser)
===========> *A first-order self-contained text* ============>
(using a prover)
==========> *A computer–made proof/verification* ===========>
(using an editor)
=======> *Text in a form comprehensible for a human*

For writing formalized (mathematical) texts, the Russian- and English-language versions of a formal language were constructed. The Russian version [10] was called

[1]Draw you attention to the fact that the Russian SAD had no name for a long time and all the references to it were made as to the system for mathematical texts processing.

Theory Language (TL), while the English version [11] — Formal Theory Language (ForTheL). Their syntactical analyzers were designed and implemented in such a way that their outputs are themselves computer internal presentations of the first-order language.

Note that TL and ForTheL reflect Glushkov's desire to have practical formal languages suitable for writing mathematical propositions and their proofs. At that, they "should relate to the existing formal languages of mathematical logic as, for example, ALGOL-60 language relates to the language of recursive functions or normal algorithms" [2].

Here we conclude the consideration of TL and ForTheL noting that there are many enough publications on these languages (the most part of such references can be found in [9]), in particular, ForTheL has its complete description in the "ForTheL Reference" located on the site "nevidal.org". In this connection, in what follows the attention is focused on logical investigations prolonging the research made in the framework of two SAD systems and oriented to the further development and improvement of methods for inference search in the EA-style.

The main EA principles relating to the proof search are: the syntactical form of a task under consideration should be preserved; logical transformations should be performed in the signature of an initial theory (i.e. without applying skolemization); proof search should be goal-oriented; equality handling/equation solving should be separated from proof searching.

After considering different known approaches to the construction of automated theorem-proving methods, one will conclude that the sequent approach is one of the most suitable for satisfying the just given principles. This is the main reason why the preference was given to the sequent formalism when constructing the Russian and English SADs. Namely, its *logical (possible) capabilities are studied in the paper from the point of view of the construction of efficient enough deductive systems for computer proof search in different first-order logics.*

Here it is important to note that the logical investigations being made for the Russian SAD, also include the investigations on resolution-type methods and interpretation of Maslov's Inverse Method. Their description has already been given in [12] in a complete enough form. This is one of the reasons why there is nothing about them in what follows.

Another reason is that the original prover of the English SAD is based on the sequent formalism only , which is caused by that the following decision was accepted when designing the English SAD: in the case of the necessity or desire to use a resolution technique, a SAD user can apply one of such well-known provers as SPASS, Vampire, E Prover, Prover9, and Otter. As a result, efforts of the paper's author as one of the SAD developers were and are directed mainly to studying the sequent formalism.

3. EA-STYLE DEDUCTION AND SEQUENT CALCULI

From the very beginning of its appearance, the EA program has paid great attention to developing machine proof search methods suitable for the various fields of mathematics and reflecting (informal) human reasoning techniques. The first attempt in this direction was made in 1963, when V. M. Glushkov formulated the problem of automated theorem proving in the group theory. In this connection, texts from the books on the group theory were exposed to the careful analysis. As a result, a machine procedure for proof search in the group theory was constructed in the middle of the 1960s [13].

Later, that procedure was generalized to the case of first-order restricted classical predicate calculus without equality [14, 15]. The procedure admitted its interpretation as a specific, sound and complete, sequent calculus later called the AGS (Auxiliary Goals Search) calculus [16]. The primary AGS satisfied the EA principles (2) and (3), where:

– for avoiding skolemization, a specific quantifiers handling technique was developed; it was actually a modification of Kanger's idea [17] about using of so-called "dummies" and "parameters" in quantifier rules as special variables with subsequent replacing "dummies" by "admissible" terms at certain instants of time and

– goal-orientation, which means that at each instant of time the succedent of any sequent under consideration should have no more than one formula-goal.

Further development of the ASG calculus was oriented to its improvement in the direction of optimizing quantifier handling, separating equality processing from deduction, and goal-driven proof searching.

The optimization of quantifier handling was achieved by the introduction of an original notion of an admissible substitution distinguished from Kanger's one.

The equality separation was oriented to the development of special methods for equality processing and equation solving. (Later algebra systems and problem solvers were suggested to use for this purpose.)

The goal-driven search was based on driving the process of an auxiliary goal generation by taking a formula (goal) under consideration into account.

All these investigations led to the construction of an original sequent calculus (with the new notion of admissibility using in this paper) published in [18] in 1981 and implemented in the (Russian) SAD system.

Additionally note that the Russian SAD contained certain tools for applying definitions and auxiliary propositions at certain time moments. They reflected the usual way of using definitions and auxiliary propositions by a mathematician and demonstrated good results in proving usual mathematical theorems.

Since then, the investigations on inference search in EA-style were stopped until 1998, when the author took a participation in the Intas project 96-0760 "Rewriting Techniques and Efficient Theorem Proving"(1998-2000), which led to the further development of the calculus from [18] in several directions. One of them was oriented to the to development of a special sequent formalism allowing to improve the positive features of AGS [19]. Another one, which found its reflection in implementing the English SAD, was in studying different ways for the construction of goal-driven sequent calculi satisfying the AGS-style proof search [20].

In this connection note that at present, more and more attention is given to elaboration of proof search methods, which are oriented to deducing in non-classical logics as well as using a man-machine interaction. However, when well-known methods relying upon the results of Skolem [21] and Herbrand [22] and having a sufficiently high efficiency (such as resolution-type methods, the inverse method, connection graph methods, etc.) are tried to use with this purpose, a number of difficulties arise. These difficulties are caused by the fact that both the specificity of the methods consisting in the "destruction" of a proposition to be proven by transforming it into clauses, assembles, connection graphs, an so on and ways of organizing of a deduction process impede the implementation of tools for the construction of such a "natural" deduction, which could be achieved by using the standard Gentzen [23] or Kanger [17] calculi.

At the same time, usual Gentzen-type sequent calculi significantly yield proof search efficiency, for example, to resolution-type methods. In general, this is mainly connected with different possible orders of the quantifier rule applications in them while resolution-type methods, due to skolemization, are free from this deficiency. That is why the contemporary state of automated reasoning is characterized by that great attention is attracted to the construction of a proof-search technique combining the best features of the first-order sequent formalism and "combinatory" logical methods (for example, such as sophisticated resolution- and paramodulation-type strategies). These words are confirmed by the great number of appropriate publications. (We refer to [24] containing such papers as [25, 26, 27, 28] and so on.)

When quantifier rules are applied in sequent calculi, a substitution of selected terms for variables is made. In order for this step of deduction to be sound, certain restrictions are put on a substitution. A substitution satisfying these restrictions is said to be admissible. Usual Gentzen (classical and intuitionistic) calculi explore Gentzen's admissibility requiring that a term substituted for the variable of a quantifier cannot contain bound variables, which proves to be sufficient for the needs of the proof theory. But it becomes useless from the point of view of the efficiency of computer-oriented inference search, since it produces extreme large step-by-step examinations of different orders of quantifier rules applications in attempting to find at least one order leading to success.

As a result, researchers in logical reasoning face with the following problem: how can the Gentzen quantifier rules be modified for providing the optimization of quantifier rules applications in sequent calculi?

The first serious attempt in this direction, seemingly, belongs to S. Kanger [17] who suggested his sequent calculus, in which a "pattern" of an inference tree is first constructed with the help of propositional rules and specific quantifier rules eliminating quantifiers and inserting only special variables, the so-called parameters and dummies. In order for the inference process to complete successfully, at some instants of time an attempt is made to convert a "pattern" into a proof tree by means of replacing each dummy in the "pattern" by a term from a list connected with the dummy. In the case of lack of success, the process is continued. But Kanger's approach still preserves the dependence of proof search efficiency on step-by-step attempts to find an order of quantifier rule applications leading to the construction of a proof tree. (Note that Kanger's approach proposed for classical logic was later incorporated into intuitionistic logic [29].)

At the end of the 1990s, there appeared two publications [30] and [31] containing the more sophisticated techniques for inference search in classical and non-classical first-order logics. They are based on the analysis of the general logical structure of formulas being investigated on deducibility and has led to the appearance of a number of deduction methods such as matrix characterization method [31], different modifications of the connection method [30] (see, for example, [32, 33, 34]), and original sequent and tableau methods (see, for example, [35, 36]).

The author's approach presented in this paper is based on [37]. It shares, in a certain sense, some ideas with [30] and [31] taking into account only the *general logical structure* of a given formula (or sequent), while, according to the author's approach, the *quantifier structure* is extracted as the most important component of the logical structure for its subsequent "analysis" with the help of the new notion of an admissible substitution. It permits to optimize the step-by-step examination of all possible orders of quantifier rule applications and leads to the construction of efficient enough methods for proof search in classical logic using only the quantifier structure of a given formula (or sequent) (see, for example, [19]). At that, it turned out that the (new) notion of admissibility is not enough for the construction of sound calculi in the intuitionistic case. This situation can be corrected by using the notion of compatibility proposed in [38] for the construction of the sound (and complete) intuitionistic tableau calculus with free variables (some historical details can be found in [12]).

This paper in a certain sense summarizes the previous author's results concerning machine proof search in classical and non-classical sequent first-order logics with the subformula and cut elimination properties. It clarifies some unclear places in the previous publications and further develops the author's approach to the construction of computer-oriented sequent calculi for proof search in the EA-style.

4. MAIN NOTIONS AND DENOTATIONS

Our research concerns the sequent form of classical and intuitionistic (modal) logics without the cut and, possibly, with equality. At that, our way of the constriction of modal calculi has a certain correlation with the papers [31, 39].

The first-order sequent terminology is used. The basic logical signature Sig_0 of our first-order language contains *logical connectives*: the *universal quantifier symbols* $\forall$ and *existential quantifier symbols* $\exists$ as well as the *propositional symbols* for the *implication* ($\supset$), *disjunction* ($\vee$), *conjunction* ($\wedge$), and *negation* ($\neg$); at that, the quantifiers $\forall x$ and $\exists x$ are often considered as a single whole.

Besides, there are a set of *functional symbols* necessarily containing a special constant, say, c_0, and a non-empty set of *predicate symbols*, containing, perhaps, the equality ($\approx$).

As a rule, the arity of any modal operator (for example, $\Box$ and $\Diamond$) is equal to 1. However, we consider a more general setting supposing that a finite set $Mod_m = \{\bigcirc_1, \ldots, \bigcirc_m\}$ of *modal operators* of any finite positive arities is given (the well-known modal operators $\Box$ and $\Diamond$ may be among them).

For a given Mod_m, the signature $Sig_0 \cup Mod_m$ is denoted by Sig_m. $Mod_0 = \emptyset$ by definition.

Extend the signature Sig_m (including the case of Sig_0 when $m = 0$) in the following way: for any natural numbers i and n ($i, n \geq 1$) and for any symbol $\odot$ from Sig_m, we add the new symbols ${}^{i}\odot$, ${}_{n}\odot$, and ${}^{i}_{n}\odot$ to Sig_m and denote this extension of Sig_m by ${}^{e}Sig_m$ (${}^{e}Sig_m = {}^{e}Sig_0 \cup {}^{e}Mod_m$); at that, i is called a *left upper index* and n a *left down index* of $\odot$.

Any symbol from ${}^{e}Sig_m$ is called a *connective*. If we want to specify that $\odot \in {}^{e}Mod_m$ ($\odot \in {}^{e}Sig_0$), we simply say that $\odot$ is a *modal* (*logical*) *connective*. Besides, the symbol $\odot$ (possibly, with upper and down indexes and right subscripts) is reserved for denoting any propositional logical symbol, quantifier, or modal operator.

For example, ${}^{1}\vee$, ${}^{5}_{7}\forall$, and ${}_{3}\Box$, may be symbols of the extended signature ${}^{e}Sig_m$; at that, ${}^{1}\vee$, ${}^{5}_{7}\forall \in {}^{e}Sig_0$ while ${}_{3}\Box \in {}^{e}Mod_m$. (Note that left upper indexes will be used for producing so-called *copies* of the same expression (see below).)

A countable set of variables used in logics under consideration is denoted by Var. At that, we consider that Var consists of two disjoint countable sets: Var_0 and ${}^{e}Var_0$ ($Var = Var_0 \cup {}^{e}Var_0$), where the following condition is satisfied: Var_0 is a set of usual variables and for any $v \in Var_0$ and any i and n ($i, n \geq 1$), ${}^{e}Var_0$ contains the *variables* ${}^{i}v$, ${}_{n}v$, and ${}^{i}_{n}v$.

The notions of a *term*, *atomic formula*, *literal*, *formula*, *subformula*, and *proper subformula* over $Sig_0 \cup Var_0$ and ${}^{e}Sig_0 \cup Var$ are used in the usual sense with the only exception: the expressions ${}^{i}_{n}\forall\, {}^{j}_{k}x$ and ${}^{i}_{n}\exists\, {}^{j}_{k}x$ are allowed to use for denoting quantifiers only in the case when upper indexes and down indexes coincide if these indexes are present (that is when $i = j$ and $n = k$).

For the case of the signature ${}^{e}Sig_m$, the notion of a formula is extended as follows:

(I) Each formula over ${}^{e}Sig_0 \cup Var$ is considered to be a formula over $Sig_m \cup Var$.

(II) If $F_1, \ldots, F_r$ are formulas over ${}^{e}Sig_m \cup Var$ and $\bigcirc$ of the arity r belongs to ${}^{e}Mod_m$, then $\bigcirc(F_1, \ldots, F_r)$ is a formula over ${}^{e}Sig_m \cup Var$.

(III) If F_1 and F_2 are formulas over ${}^{e}Sig_m \cup Var$, $\odot \in {}^{e}Sig_m$ is a logical connective with arity 2 (i.e. $\supset$, $\vee$, or $\wedge$, maybe, with left upper and down indexes), and $\odot'$ is a logical connective with arity 1 (i.e. $\exists x$, $\forall x$, or $\neg$, maybe, with left upper and down indexes), then $F_1 \odot F_2$ and $\odot' F_1$ is a formula over ${}^{e}Sig_m$.

Let a formula F over ${}^{e}Sig_m \cup Var$ be of the form $\odot(F_1, \ldots, F_r)$, $\odot F_1$, or $F_1 \odot F_2$, where $\odot$ is a modal operator or a logical symbol. Then $\odot$ is called a *principal symbol* (*connective*) of F and $F_1, F_2, \ldots, F_r$ are called *principal subformulas* of the formula F, which sometimes will be called a $\odot$*-formula*.

The notions of *free* and *bound variables* of a formula and its *variant* (over both ${}^{e}Sig_0 \cup Var$ and ${}^{e}Sig_m \cup Var$) are usual.

Sequents are defined in the usual manner, i.e. as an expression of the form $F_1, \ldots, F_p \to G_1, \ldots, G_q$, where $F_1, \ldots, F_p$, $G_1, \ldots, G_q$ are formulas over ${}^{e}Sig_m \cup Var$ except that the succedent and antecedent of a sequent are considered as finite multisets (see, for example, [39] and [42]).

A *tree* is understood in the usual sense. A tree with nodes labeled by sequents is called a *sequent tree*.

Any syntactical object over $Sig_m \cup Var_0$ (including variables, terms, formulas, sequents, and so on) is called an *original* expression.

As usual, the following *convention* holds w.r.t. bound variables: two different quantifiers in any formula or sequent over ${}^{e}Sig_m \cup Var$ cannot have common variable; moreover, any formula or sequent cannot simultaneously contain any variable being both bound and free. (It is known that these conditions can be achieved by means of renaming bound variables, which does not effect on deducibility).

A formula without free variables is called *closed*.

For solving the problem of deducibility in a usual sequent calculus, we can restrict us, without loss of generality, by establishing the deducubility of only an *initial sequent* of the form $\to F$, where F is a closed original formula.

In what follows, the original formula $F^* = \exists y \neg \Box \exists x P(x, f(y)) \supset \neg \forall y' \Box \exists x' P(x', y')$ (P is a predicate symbol and f a functional symbol) deducible in **GK** [39] will be used in a number of examples clarifying the notions introduced and results obtained.

Any expression over ${}^{e}Sig_m \cup Var$, all connectives and variables of which contain left upper indexes, is called (upper) *indexed*.

Note that according to this definition, *predicate and functional symbols cannot be indexed*.

The result of any upper indexing of an expression without upper indexes (i.e. adding left upper indexes to all connectives and variables of the expression) is called its *copy*. Two copies of the same expression (including the case of a variable or connective) are considered to be *copies of each other* by definition. That is the relation "to be a copy" is transitive.

The extension of all the necessary semantic notions to formulas and sequents containing upper and down left indexes is obvious: it is enough to omit all their indexes and use the usual semantic notions.

Remark. In the case of $Mod_0(=\emptyset)$, all the original formulas and sequents are those syntactical units that are used when determining the standard classical and intuitionistic calculi. The case of $Mod_1 = \{\Box\}$ leads to different modal sequent calculi depending on what sequent rules for $\Box$ are determined; for example, there exists the possibility to determine **GK** or **GS4** from [39].

A *substitution* is a finite mapping from variables to terms presented in the form $\{x_1 \mapsto t_1, \ldots, x_n \mapsto t_n\}$, where $x_1, \ldots, x_n$ are pairwise distinct variables and each term t_i is not x_i for any $i = 1, \ldots, n$; at that, each $x_i \mapsto t_i$ is called a *substitution component*, t_i a *term*, and x_i a *variable* of σ $(i = 1, \ldots, n)$ (cf. [40]).

A *renaming* is a substitution not containing terms distinguished from variables and replacing different variables by different ones.

For a substitution σ and quantifier-free expression Ex (including the cases of a formula and sequent), the result of the application of σ to Ex is understood in the usual sense; it is denoted by $Ex \cdot \sigma$.

If σ and λ are substitutions, then $\sigma \cdot \lambda$ denotes their *composition*, i.e. a substitution, the result of the application of which to Ex is equal to $(Ex \cdot \sigma) \cdot \lambda$.

The notions of a *unifier*, *simultaneous unifier*, and *the most general simultaneous unifier* of a set of expressions are understood in the usual sense [40].

For formulas F and G, we understand the notions of *positive* $(G\lfloor F^+ \rfloor)$ and *negative* $(G\lfloor F^- \rfloor)$ occurrences of F in G in the sense of the paper [41].

More precisely, let a formula F have one or more occurrences in a formula G. Let us fix a certain occurrence of F in G. This occurrence is called *positive* (*negative*) in the accordance with the following:

- ∘ $G\lfloor F^+ \rfloor$, if F coincides with G;
- ∘ $G\lfloor F^+ \rfloor$ $(G_1\lfloor F^- \rfloor)$, if G is of the form: $G_1 \wedge G_2$, $G_2 \wedge G_1$, $G_1 \vee G_2$, $G_2 \vee G_1$, $G_2 \supset G_1$, $\forall x G_1$, or $\exists x G_1$ and $G_1\lfloor F^+ \rfloor$ $(G_1\lfloor F^- \rfloor)$;
- ∘ $G\lfloor F^- \rfloor$ $(G\lfloor F^+ \rfloor)$, if G is of the form: $G_1 \supset G_2$ or $\neg G_1$ and $G_1\lfloor F^+ \rfloor$ $(G_1\lfloor F^- \rfloor)$;
- ∘ $G\lfloor F^+ \rfloor$ $(G\lfloor F^- \rfloor)$, if G is of the form $\bigcirc(G_1, \ldots, G_r)$ and $G_i\lfloor F^+ \rfloor$ $(G_i\lfloor F^- \rfloor)$ $(1 \le i \le r)$, where $\bigcirc$ is a modal operator.

Moreover (cf. [41]), a selected occurrence of a formula in a sequent $\Gamma \to \Delta$ is called a *positive* (*negative*) one if, and only if, this occurrence is a positive one in a formula from Δ (from Γ) or a negative one in a formula from Γ (from Δ).

If a formula F is of the form $\forall x F'$ $(\exists x F')$ and F has a positive (negative) occurrence in a formula G or a sequent S, then $\forall x$ $(\exists x)$ is called a *positive quantifier* in G or S respectively; $\exists x$ $(\forall x)$ is called a *negative quantifier* in G or S, if $\exists x F'$

$(\forall x F')$ has a positive (negative) occurrence in G or S respectively.

As the convention about bound variables is satisfied, any quantifier cannot have more then one (positive or negative) occurrence in a formula or sequent under consideration. So the following definitions (cf. [17]) do not lead to misunderstanding.

Following [17], the variable of a positive quantifier occurring in a formula G or sequent S is called a *parameter* in G or S respectively; the variable of a negative quantifier occurring in G or S is called a *dummy* in G or S respectively.

Note that we *preserve* the name "variable" for denoting both dummies and parameters in the cases, when it is only important that they occur in a formula or sequent under consideration.

For F^*, we have: x and y' are its dummies and x' and y are its parameters.

Since the properties "to be a dummy" and "to be a parameter" are invariant for any variable x w.r.t. any rules applications in our sequent calculi, it turns out convenient to write $\overline{x}$ in order to indicate that x is a parameter and to write $\underline{x}$ in order to indicate that x is a dummy in a formula or sequent under consideration. This convention will be used often by default in what follows.

For a formula F (sequent S), $\mu(F)$ ($\mu(S)$) denotes the result of the removing of all the quantifiers from F (from S).

The operation μ has the following obvious properties w.r.t. logical and modal connectives (below F_1, ..., F_r are formulas, x a variable, $\bigcirc$ a modal operator):

$\mu(\neg F_1) = \neg\mu(F_1)$, $\mu(F_1 \wedge F_2) = \mu(F_1) \wedge \mu(F_2)$, $\mu(F_1 \vee F_2) = \mu(F_1) \vee \mu(F_2)$, $\mu(\forall x F_1) = \mu(F_1)$, $\mu(\exists x F_1) = \mu(F_1)$, and $\mu(\bigcirc(F_1, \ldots, F_r)) = \bigcirc(\mu(F_1), \ldots, \mu(F_r))$.

If F (S) is a formula (sequent) and x its parameter or dummy, then x is considered as a parameter or dummy in $\mu(F)$ (in $\mu(S)$).

Extend the operation of the application of a substitution σ to a quantifier-free expression on the case of the application of σ to any formula F in the following way (denoting the result by $F \cdot \sigma$): produce $\mu(F)$ (by omitting all quantifiers in F), construct $\mu(F) \cdot \sigma$, and restore all the omitted qualifiers at their initial places.

For example, for F^* and $\sigma = \{x \mapsto g(x,y), y' \mapsto c\}$, we have that $F^* \cdot \sigma = \exists y \neg \Box \exists x P(g(x,y), f(y)) \supset \neg \forall y' \Box \exists x' P(x', c)$.

The extension of the operation "$\cdot$" on sequents and sequent trees is obvious. Note that the result of its application to a sequent or sequent tree produces a sequent or sequent tree respectively.

We use the usual definition of a *sequent calculus* while the deduction of a sequent in it has the form of an *inference tree* growing "from top to bottom" according to (counter-)applications of inference rules "from top to bottom". An inference tree is called a *proof tree*, if all its leaves are labeled by axioms. A sequent S is *deducible* in a calculus, if there exist a proof tree for S in the calculus.

Note that our sequent calculi do not contain the cut rule as the cut is supposed to be *eliminable*. Becides, all their rules satisfies the usual *subformula property* (see, for example, [23] or [42]).

5. KANGER-TYPE CALCULI WITH USUAL QUANTIFIER RULES

All our first-order sequent calculi are based on certain modifications of Gentzen's calculi **LK** and **LJ** without equality from [23]; at that, $\mathbf{LK}^{\approx}$ and $\mathbf{LJ}^{\approx}$ will denote **LK** and **LJ** containing the equality rules in Kanger's form from [17]. The consideration of the succedents and antecedents of sequents as multisets allows us to convert the calculi **LK** and **LJ** into **KK** and **KJ** by removing the weakening and exchange rules from them. After adding the equality rules in the form from [17] to **KK** and **KJ**, we get the calculi $\mathbf{KK}^{\approx}$ and $\mathbf{KJ}^{\approx}$.

Note that in these notations, Kanger's calculus from [17] can be considered as coinciding with $\mathbf{KK}^{\approx}$, provided that in all the rules, the left upper indexes and subscribes are absent. This calculus is denoted by $\mathbf{K}^{\approx}$; $\mathbf{K}^{\approx}$ without the equality rules is denoted by **K**.

All the just described calculi are presented in Fig. 1. They form a calculi family denoted by **KC** containing all the rules and axioms from Fig. 1 except $(Con^* \rightarrow)$, $(\rightarrow Con^*)$, $(\exists^* \rightarrow)$, and $(\rightarrow \exists^*)$. Different its modifications lead to the family, the most general representative of which is denoted by $\mathbf{KK}^{\approx}$. That is why we attempt to obtain certain results for **KC** and then modify them in a certain way.

For introducing the modal logics, we follow the paper [39], where necessary modal rules are simply added to Gentsen's calculi **LK** and **LJ**. Doing the same for a set $\mathbf{Mod}_m$ consisting of m modal rules, we define a family $\mathbf{KK}$+$\mathbf{Mod}_m$ containing the following classes of modal logics: $\mathbf{KK}$+$\mathbf{Mod}_m$, $\mathbf{KJ}$+$\mathbf{Mod}_m$, $\mathbf{KK}^{\approx}$+$\mathbf{Mod}_m$, and $\mathbf{KJ}^{\approx}$+$\mathbf{Mod}_m$. The same concerns the modal extensions of **K** and $\mathbf{K}^{\approx}$.

As to modal rules, we consider that any modal rule (M_l) from $\mathbf{Mod_m}$ $(1 \leq l \leq m)$ is of the form:

$$\frac{\Gamma, {}^{i_{l,1}}_{n_{l,1}}\bigcirc_{l,1}(\Phi_{l,1}), \ldots, {}^{i_{k_{l,k_l}}}_{n_{l,k_l}}\bigcirc_{l,k_l}(\Phi_{l,k_l}) \rightarrow {}^{j_{l,1}}_{r_{l,1}}\bigcirc'_{l,1}(\Psi_{l,1}), \ldots, {}^{j_{l,k'_l}}_{r_{l,k'_l}}\bigcirc'_{l,k'_l}(\Psi_{l,k'_l}), \Delta}{\Gamma, \Phi_{l,1}, \ldots, \Phi_{l,k_l} \rightarrow \Psi_{l,1}, \ldots, \Psi_{l,k'_l}, \Delta},$$

where $\Phi_{l,1}, \ldots, \Phi_{l,k_l}$ and $\Psi_{l,1}, \ldots, \Psi_{l,k'_l}$ are multisets of formulas.

Let us introduce so-called *numbered* formulas having a great importance for our considerations and being used in the description of the calculi from Fig. 1.

Let F be an original formula (i.e. a formula over over $Sig_m \cup Var_0$) being different from an atomic formula. By (n, F) denote the *occurrence* of the nth subformula of F when reading the formula F from left to right (n does not exceed the number of connectives occurring in F).

If (n, F) is an occurrence of a $\odot$-formula in a formula F, then the connective $\odot$ is said to be the ${}_n\odot$-*connective* in F; at that, the $\odot$-formula is called a ${}_n\odot$-*subformula* of F, where n is explicitly indicated as a left down index of ${}_n\odot$. In the case if $\odot$ is $\forall x$ or $\exists x$, ${}_n\odot$ is ${}_n\forall_n x$ or ${}_n\exists_n x$ respectively and all the other occurrences of x in F receive n as their left down index.

This operation of the left down indexing of all connectives and variables in an original formula F is called the *numbering* operation and the result of its application to F is called a *numbered formula* and denoted by ${}_\nu F$.

For the formula F^*, we have the following result ${}_\nu F^*$ of numbering its connectives and variables: ${}_1\exists_1 y \; {}_2\neg \; {}_3\Box \; {}_4\exists_4 x \; P({}_4x, f({}_1y)) \; {}_5\supset \; {}_6\neg \; {}_7\forall_7 y' \; {}_8\Box \; {}_9\exists_9 x' \; P({}_9x', {}_7y')$ (pay your attention that P and f have no indexes).

Note that due to the numbering of the connectives in an original formula F, all these connectives become *graphically pairwise different* ones. At that, we consider that the logical interpretation of the numbered formula F coincides with the interpretation of F without left (upper and down) indexes.

Let F be an original formula. If ${}_\nu F$ contains ${}_m\odot$- and ${}_n\odot'$-subformulas $(m \neq n)$ and the ${}_m\odot$-subformula is a subformula of the ${}_n\odot'$-subformula, then ${}_n\odot'$ and ${}_m\odot$ are said *to be in the relation* $\lessdot_F$ and this fact is denoted by ${}_n\odot' \lessdot_F {}_m\odot$.

Propositional rules :

$$\frac{\Gamma, A \,{}^{i}_{n}\!\wedge B \to \Delta}{\Gamma, A, B \to \Delta}\ (\wedge \to) \qquad \frac{\Gamma \to A \,{}^{i}_{n}\!\wedge B, \Delta}{\Gamma \to A, \Delta \quad {}^{l+1}[\Gamma] \to {}^{l+1}[B], {}^{l+1}[\Delta]}\ (\to \wedge)$$

$$\frac{\Gamma, A \,{}^{i}_{n}\!\vee B \to \Delta}{\Gamma, A \to \Delta \quad {}^{l+1}[\Gamma], {}^{l+1}[B] \to {}^{l+1}[\Delta]}\ (\vee \to)$$

$$\frac{\Gamma \to A \,{}^{i}_{n}\!\vee B, \Delta}{\Gamma \to A, \Delta}\ (\to \vee_1) \qquad \frac{\Gamma \to A \,{}^{i}_{n}\!\vee B, \Delta}{\Gamma \to B, \Delta}\ (\to \vee_2)$$

$$\frac{\Gamma, A \,{}^{i}_{n} \supset B \to \Delta}{\Gamma, A \,{}^{i}_{n} \supset B \to A, \Delta \quad {}^{l+1}[\Gamma], {}^{l+1}[B] \to {}^{l+1}[\Delta]}\ (\supset\to) \qquad \frac{\Gamma \to A \,{}^{i}_{n} \supset B, \Delta}{\Gamma, A \to B, \Delta}\ (\to\supset)$$

$$\frac{\Gamma, \,{}^{i}_{n}\neg A \to \Delta}{\Gamma \to A, \Delta}\ (\neg \to) \qquad \frac{\Gamma \to {}^{i}_{n}\neg A, \Delta}{\Gamma, A \to \Delta}\ (\to \neg)$$

Contraction rules :

$$\frac{\Gamma, A \to \Delta}{\Gamma, {}^{l+1}[A], A \to \Delta}\ (Con \to) \qquad \frac{\Gamma \to A, \Delta}{\Gamma \to A, {}^{l+1}[A], \Delta}\ (\to Con)$$

$$\frac{\Gamma, A \to \Delta}{\Gamma, {}^{i}_{n}{}^{x<(l+1)}[A], A \to \Delta}\ (Con^{\#} \to) \qquad \frac{\Gamma \to A, \Delta}{\Gamma \to A, {}^{i}_{n}{}^{x<(l+1)}[A], \Delta}\ (\to Con^{\#})$$

Quantifier rules :

$$\frac{\Gamma \to {}^{i}_{n}\forall^{i}_{n} xA, \Delta}{\Gamma \to {}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\overline{x}}, \Delta}\ (\to \forall) \qquad \frac{\Gamma, {}^{i}_{n}\forall^{i}_{n} xA \to \Delta}{\Gamma, A|^{{}^{i}_{n}x}_{t_i} \to \Delta}\ (\forall \to) \qquad \frac{\Gamma, {}^{i}_{n}\forall^{i}_{n} xA \to \Delta}{\Gamma, {}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\underline{x}} \to \Delta}\ (\forall^{\#} \to)$$

$$\frac{\Gamma, {}^{i}_{n}\exists^{i}_{n} xA \to \Delta}{\Gamma, {}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\overline{x}} \to \Delta}\ (\exists \to) \qquad \frac{\Gamma \to {}^{i}_{n}\exists^{i}_{n} xA, \Delta}{\Gamma \to A|^{{}^{i}_{n}x}_{t_i}, \Delta}\ (\to \exists) \qquad \frac{\Gamma \to {}^{i}_{n}\exists^{i}_{n} xA, \Delta}{\Gamma \to {}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\underline{x}}, \Delta}\ (\to \exists^{\#})$$

Equality rules :

$$\frac{\Gamma,\ t' \approx t'' \to \Delta}{\Gamma|^{t'}_{t''},\ t' \approx t'' \to \Delta|^{t'}_{t''}}\ (\approx_1\to) \qquad \frac{\Gamma,\ t'' \approx t' \to \Delta}{\Gamma|^{t'}_{t''},\ t'' \approx t' \to \Delta|^{t'}_{t''}}\ (\approx_2\to)$$

Axioms :

$$\underline{\Gamma, A \to A, \Delta} \qquad \underline{\Gamma \to t \approx t,\ \Delta}$$

KC contains all the rules except $(\forall^{\#} \to)$, $(\to \exists^{\#})$, $(Con^{\#} \to)$, and $(\to Con^{\#})$. **BC** contains all the rules except $(\forall \to)$, $(\to \exists)$, $(Con^{\#} \to)$, and $(\to Con^{\#})$. **QC** contains all the rules except all the quantifier rules. In (Ax), A is an atomic formula. The number l denotes the largest left upper index in a just-constructed inference tree. In $(\to \forall)$ and $(\exists \to)$, ${}^{(l+1)+}{}^{i}_{n}\overline{x}$ is a (new) parameter, that is the eigenvariable condition is satisfies for it. In $(\forall^{\#} \to)$ and $(\to \exists^{\#})$, ${}^{(l+1)+}{}^{i}_{n}\underline{x}$ is a (new) dummy. The terms t, t', and t'' do not contain dummies; moreover, besides parameters and c_0, all these terms may contain constants and functional symbols occurring in only an initial sequent. $\Gamma|^{t'}_{t''}$ and $\Delta|^{t'}_{t''}$ are the results of the replacement of all the occurrences of t' by t''. The same holds for ${}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\underline{x}}$, ${}^{l+1}[A]|^{{}^{i}_{n}x}_{(l+1)+{}^{i}_{n}\overline{x}}$, and $A|^{{}^{i}_{n}x}_{t_i}$, where the term t_i is free for ${}^{i}_{n}x$ in A. The case of *classical* (modal) logics takes place, if there is no restrictions for sequents. The case of *intuitionistic* (modal) logics takes place, if the succedent of any sequent in any inference rule contains no more than one formula; at that, in $(\to Con)$ the sequent lying under the line is of the form $\Gamma \to {}^{l+1}[A]$ and in $(\to Con^{*})$ the sequent lying under the line is of the form $\Gamma \to {}^{x<(l+1)}[A]$.

Fig. 1. Calculi **KC**, **BC**, and **QC**

We also write ${}_n\odot' \lessdot_F {}_mx$ (${}_ny \lessdot_F {}_m\odot$), when ${}_m\odot$ (${}_n\odot'$) is ${}_m\forall_m x$ or ${}_m\exists_m x$ (${}_n\forall_n y$ or ${}_n\exists_n y$). In the case when ${}_m\odot$ and ${}_n\odot'$ simultaneously are quantifiers of one of the just given forms, we simply write ${}_ny \prec_F {}_mx$ determining the relation $\prec_F$ and underlining the fact that $\lessdot_F$ *is restricted to the case of quantifier variables.*

Further, by definition, ${}_n\odot' \lessdot_F {}_m\odot$ implies that ${}^i_n\odot' \lessdot_F {}^j_m\odot$ and ${}_ny \prec_F {}_mx$ implies ${}^i_ny \prec_F {}^j_mx$ for any i and j.

For the formula F^* numbered above, we have for example: ${}_5\supset$ is the least element of $\lessdot_{F^*}$ and, for example, ${}_1y \prec_{F^*} {}_4x$, ${}_2\neg \lessdot_{F^*} {}_4x$, ${}_6\neg \lessdot_{F^*} {}_8\Box$; the variables x and x' are not comparable, the same concerns ${}_4x$ and ${}_6\neg$ as well as ${}_3\Box$ and ${}_8\Box$.

Obviously, for any numbered original formula F, the *relations* $\prec_F$ and $\lessdot_F$ are *irreflexive, asymmetric, and transitive*, that is they are *strict orders* over the sets ${}^eVar$ and ${}^eSig_m$ respectively.

It is also evident that these orders fully determined by an initial closed formula F remain unchanged during the deduction process in any our calculi. That is why there will not be any contradiction and misunderstanding in using such denotations as $\prec_S$ and $\lessdot_S$ for an (indexed) sequent S, $\prec_\Xi$ and $\lessdot_\Xi$ for a set Ξ of (indexed) formulas, and $\prec_{Tr}$ and $\lessdot_{Tr}$ for an (indexed) sequent tree Tr.

If G distinguished from an atomic formula is a $\odot$-copy of a proper subformula of a (numbered) initial formula F, then both $\odot$ and any connective $\odot'$ from G such as $\odot \lessdot_F \odot'$ (if it exists) are called *in-connectives* (*inner connectives*) for G w.r.t. F. If at that time $\odot$ ($\odot'$) are $\forall x$ or $\exists x$, x is called an *in-variable* (*inner variable*) for G w.r.t. F.

By definition, any atomic formula is considered as *not containing* inner connectives and variables, while all the connectives and variables of any original (*closed*) formula F are considered to be *only inner* for F w.r.t. F.

If G is a subformula of a numbered original formula F, then both any connective and any variable not being inner for G w.r.t. F are called, respectively, an *ex-connective* (*external connective*) and *ex-variable* (*external variable*) for G w.r.t. F.

This definition implies that any original formula *does not contain* any external variables w.r.t. itself.

Let G be a formula received from a copy of a subformula G' of an original formula F by removing some or all quantifiers. If k is a natural number, then the formula ${}^k_F[G]$ is constructed in the following way ($k \geq 1$): for any in-connective (in-variable) for G w.r.t. F, i is replaced by $k+i$ in G provided that i is the left upper index of this in-connective (in-variable); if there is no index in this in-connective (in-variable), k becomes its left upper index in ${}^k_F[G]$.

When F is known or there is no matter what F denotes, we simply write ${}^k[G]$. (This denotation is used in Fig. 1.)

If Γ is a multiset of copies of subformulas of a formula F, then ${}^k_F[\Gamma] = \{{}^k_F[H] : H \in \Gamma\}$. For a sequent S of the form $\Gamma \to \Delta$, ${}^k_F[S]$ denotes the sequent ${}^k_F[\Gamma] \to {}^k_F[\Delta]$, where Γ and Δ are multisets of copies of subformulas of a formula F.

Note that the operation ${}^k_F[\cdot]$ transforms any graphically different connectives and variables into graphically different ones renaming parameters into only (new) parameters and dummies into only (new) dummies.

All our sequent calculi are intended for the establishing of the deducibility of an *initial* sequent of the form $\to F$ (F is a closed initial formula) in a standard sequent calculus containing usual propositional, modal, and quantifier rules, to which the equality rules from [17] can be added, if necessary. To do this, we always make attempt to construct a proof tree for a sequent of the form $\rightarrow {}^1[{}_\nu F]$ called a *starting sequent* in our sequent calculi ${}_\nu F$ is a numbered formula).

Remark about numbered formulas. Let $\rightarrow F$ be an initial sequent and ${}_\nu F$ is a numbered original formula, containing a ${}_n\odot$-formula. Then the inference rules from Fig. 1 show that the number n always is an invariant w.r.t. any syntactical and logical transformations that can be performed in any calculus under consideration. This means that n is a left down index of only a copy of the symbol ${}_n\odot$ appeared in making deduction from $\rightarrow {}^1[{}_\nu F]$ in a given calculus.

As the calculi under consideration possess the *subformula property*, this will lead to producing instants of subformulas of ${}^1[{}_\nu F]$, their variants, and/or copies only.

The establishing of the deducibility of any starting sequent in $\mathbf{KC}+\mathbf{Mod}_m$ *always is made in the Kanger style*, which weans the following.

In *the first step*, only quantifier rules are applied in any order as many times as possible. In *the second step*, we apply propositional, contraction, and/or modal rules in any order as long as there is no possibility for applying a quantifier rule or stopping such applications for going to the third stage. In *the third step*, in the case of necessity or on the basis of some reason, we are applying only equality rules trying to construct a proof tree; if such attempts are unsuccessful, we omit this step and return to the first step.

An inference tree being the result of performing first two first steps of proof search is called an *equality-independent* inference tree. The one being the result of performing three steps is called an *equality-dependent* inference tree. Correspondingly, two first steps of the construction of an inference trees is called *equality-independent stage* and the third step is called an *equality-dependent* one.

Additionally note that $\mathbf{KC}+\mathbf{Mod}_m$ was defined in such a way that the following *important property* takes place: any two nodes of any inference tree in a calculus under consideration, not lying in the same branch, have no common bound variables. This property will often be used implicitly.

Comparing $\mathbf{KK}^{\approx}$ with $\mathbf{K}^{\approx}$, we obtain the following result.

Proposition 1. *An initial usual sequent $\rightarrow F$ is deducible in $\mathbf{K}^{\approx}$ ($\mathbf{K}$) if, and only if, the starting sequent $\rightarrow {}^1[{}_\nu F]$ is deducible in $\mathbf{KK}^{\approx}$ ($\mathbf{KK}$).*

Proof. If Tr is a proof tree for $\rightarrow {}^1[{}_\nu F]$ in $\mathbf{KK}^{\approx}$ ($\mathbf{KK}$), then after removing all the left upper and down indexes in it, the tree Tr will be transformed into a proof tree Tr' for $\rightarrow F$ in $\mathbf{K}^{\approx}$ ($\mathbf{K}$).

In the opposite direction. Let Tr' be a proof tree for $\rightarrow F$ in $\mathbf{K}^{\approx}$ ($\mathbf{K}$). Then we can replace $\rightarrow F$ by $\rightarrow {}^1[{}_\nu F]$ at the root of Tr' and, after this, each application of inference rule of $\mathbf{K}^{\approx}$ ($\mathbf{K}$) in Tr' can be replaced by the application of the corresponding inference rule of $\mathbf{KK}^{\approx}$ ($\mathbf{KK}$) when looking through Tr' from top to bottom and left to right, finally producing a tree Tr. Obviously, Tr is a proof tree for $\rightarrow {}^1[{}_\nu F]$ in $\mathbf{KK}^{\approx}$ ($\mathbf{KK}$). □

Remark. Proposition 1 gives the possibility to obtain the results about the soundness and completeness of $\mathbf{KK}^{\approx}$ and $\mathbf{KK}$. Therefore, the same results can take place for $\mathbf{KK}^{\approx}+\mathbf{Mod}_m$ and $\mathbf{KK}+\mathbf{Mod}_m$ in the case the soundness and completeness of the calculi $\mathbf{K}^{\approx}+\mathbf{Mod}_m$ and $\mathbf{K}+\mathbf{Mod}_m$.

6. ADMISSIBILITY AND COMPATIBILITY

When trying to implement any of the calculi from $\mathbf{KC}$+$\mathbf{Mod}_m$ on computer, one faces the problem of constructing an efficient technique for looking for an order of inference rule applications, leading to success, among all possible orders of inference rule applications. And although for the case of propositional rules there are good enough methods for solving this problem (in particular, the goal-oriented technique), in the case of quantifier rules satisfying the Gentzen admissibility, they are not practically working as the Gentzen admissibility says nothing essential about a preferable order of quantifier rule applications. Likely, this fact forced researchers in automated theorem proving to use skolemization for classical logic (cf. [43]) as the best way for solving the just mentioned problem. Likely, this coursed the appearance of investigations, in which a technique based on the sophisticated usage of Skolem functions was developed for logics not allowing the preliminary skolemization (see, e.g., [44] and [45]).

Beside of skolemization, one can try to use Kanger's technique for the optimization of quantifier rule applications by using so-called "dummies" and "parameters" and splitting the process of an inference search into Kanger's stages, which leads to the Kanger notion of admissibility of a substitution. This notion is more efficient than Gentzen's one and can be used in the case of both classical [17] and intuitionistic [29] logics.

However, the Kanger admissibility still does not allow to attain the efficiency comparable with that is observed when the preliminary skolemization is made. This is observed due to the fact that, as in the case of the Gentzen admissibility, it is required to select a certain order of the quantifier rule applications and, if it proves to be unsuccessful, another order of applications should be tried, and so on.

This was one of the reasons for the paper's author to make an attempt to eliminate this obstacle by means of introducing a new notion of the admissibility of a substitution that takes into account only the quantifier structure of formulas in an initial sequent. At last, in the late 1970s, the new notion of admissibility being described below was proposed by the paper's author. It first was announced in [37] where it was used for the construction of an Herbrand theorem without preliminary skolemization for formulas of classical logic being in prenex form. Later, this result was generalized on the case of arbitrary formulas of classical logic [46].

Note that for the case of classical logic and some of its modal extensions, the new admissibility is sufficient for the soundness of deduction in some calculi [19]. But in the case of, for example, intuitionistic logic, the usage of only admissibility cannot guarantee the soundness of the construction of a proof tree [47] because it is still necessary to take into account the fact that there are certain requirements either to a form of produced sequents or a method of their processing in inference trees. (For example, it is often required in the intuitionistic case, that the succedents of initial and inferred sequents contain no more than one formula.) That is, in such calculi, admissibility should be correlated in some way with the process of constructing inference trees. Namely for providing such correlation, the notion of the compatibility of an inference tree with a substitution is introduced below.

For any formula F (sequent S, set Ξ of formulas or sequents, sequent tree Tr), each substitution σ induces a (possibly, empty) relation $\ll_{F,\sigma}$ ($\ll_{S,\sigma}, \ll_{\Xi,\sigma}, \ll_{Tr,\sigma}$)

over the set of variables of F (S, Ξ, Tr) as follows: $y \ll_{F,\sigma} x$ ($y \ll_{S,\sigma} x$, $y \ll_{\Xi,\sigma} x$, $y \ll_{Tr,\sigma} x$) if, and only if, there exists $x \mapsto t \in \sigma$ such that x is a dummy in F (S, Ξ, Tr) and t a term containing y being a parameter in F (S, Ξ, Tr). Obviously, $\ll_{F,\sigma}$ ($\ll_{S,\sigma}$, $\ll_{\Xi,\sigma}$, $\ll_{Tr,\sigma}$) is irreflexive and transitive.

For example, let us consider the substitution $\sigma^* = \{x \mapsto x', y' \mapsto f(y)\}$, where x and y' are dummies and x' and y parameters in F^*. Then $x' \ll_{F^*,\sigma^*} x$ and $y \ll_{F^*,\sigma^*} y'$.

In what follows, for a substitution σ and formula F (sequent S, set Ξ of formulas or sequents, sequent tree Tr), $\lhd_{F,\sigma}$ ($\lhd_{S,\sigma}$, $\lhd_{\Xi,\sigma}$, $\lhd_{Tr,\sigma}$) denotes the transitive closure of $\prec_F \cup \ll_{F,\sigma}$ ($\prec_S \cup \ll_{S,\sigma}$, $\prec_\Xi \cup \ll_{\Xi,\sigma}$, $\prec_{Tr} \cup \ll_{Tr,\sigma}$). Analogously, $\blacktriangleleft_{F,\sigma}$ ($\blacktriangleleft_{S,\sigma}$, $\blacktriangleleft_{\Xi,\sigma}$, $\blacktriangleleft_{Tr,\sigma}$) denotes the transitive closure of $\lessdot_F \cup \ll_{F,\sigma}$ ($\lessdot_S \cup \ll_{S,\sigma}$, $\lessdot_\Xi \cup \ll_{\Xi,\sigma}$, $\lessdot_{Tr} \cup \ll_{Tr,\sigma}$).

Obviously, $\lhd_{F,\sigma} \subseteq \blacktriangleleft_{F,\sigma}$ ($\lhd_{S,\sigma} \subseteq \blacktriangleleft_{S,\sigma}$, $\lhd_{\Xi,\sigma} \subseteq \blacktriangleleft_{\Xi,\sigma}$, $\lhd_{Tr,\sigma} \subseteq \blacktriangleleft_{Tr,\sigma}$).

A substitution σ is called *admissible* (cf. [29, 31]) for a formula F (sequent S, set Ξ of formulas or sequents, sequent tree Tr) if, and only if, for every $x \mapsto t \in \sigma$, x is a dummy in F (S, Ξ, Tr) and $\lhd_{F,\sigma}$ ($\lhd_{S,\sigma}$ $\lhd_{\Xi,\sigma}$, $\lhd_{Tr,\sigma}$) is an irreflexive relation.

For the above-given formula F^* and substitution σ^*, we have: $\lhd_{F^*,\sigma^*} = \{\langle y, y'\rangle, \langle y, x'\rangle, \langle y, x\rangle, \langle y', x'\rangle, \langle y', x\rangle, \langle x', x\rangle\}$. Thus, σ^* is admissible substitution for F^*.

If $\sigma' = \{y' \mapsto x'\}$, then $x' \ll_{\sigma'} y'$. Since $y' \prec x'$, we have $\langle x', x'\rangle \in \lhd_{F^*,\sigma'}$. Therefore, $\lhd_{F^*,\sigma'}$ is not irreflexive and σ' is not admissible for F^*.

Let **SC** denote a Gentzen-type sequent calculus based on the usual notion of a sequent and intended for the establishing of deducibility in a logic under consideration. Suppose Tr is an inference tree in **SC** for a numbered (original or starting) sequent S and ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ a sequence of all the connectives being eliminated in Tr when applying inference rules. Let $\pi_{Tr}({}_{j_i}\odot_i)$ denote an inference rule application eliminating ${}_{j_i}\odot_i$. If Tr can be constructed according to the order determined by the sequence $\pi_{Tr}({}_{j_1}\odot_1), \ldots, \pi_{Tr}({}_{j_r}\odot_r)$, then ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ is called a *proper sequence* for Tr_F w.r.t. **SC** for S. (Obviously, there may exist a connectives occurrences sequence, being not proper for Tr.)

An inference tree Tr in **SC** is called *compatible* with a substitution σ *w.r.t.* **SC** if, and only if, there exists a proper sequence ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ for Tr such that for any natural numbers m and n, the inequality $m < n$ implies that the ordered pair $\langle {}_{j_n}\odot_n, {}_{j_m}\odot_m\rangle$ *does not belong* to $\blacktriangleleft_{Tr,\sigma}$; at that, ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ is called a *sequence providing the compatibility* of Tr with σ in **SC**.

The following proposition demonstrates the connection of admissibility with compatibility.

LEMMA 1. *Let Tr be an inference tree for a starting sequent $\rightarrow {}^1[_\nu F]$ in a* **SC** *calculus and σ a substitution. The relation $\blacktriangleleft_{Tr,\sigma}$ is irreflexive if, and only if, so is $\lhd_{Tr,\sigma}$.*

PROOF. (=>) Since $\lhd_{Tr,\sigma} \subseteq \blacktriangleleft_{Tr,\sigma}$, the necessity is obvious.

(<=) Reductio ad absurdum. Assume $\lhd_{Tr,\sigma}$ is an irreflexive relation while $\blacktriangleleft_{Tr,\sigma}$ is not. This means that $\langle \odot, \odot\rangle \in \blacktriangleleft_{Tr,\sigma}$ for a connective $\odot$ distinguished from a variable. The relation $\lessdot_{Tr}$ is irreflexive, it coincides with its transitive closer, and $\lessdot_{Tr} \subseteq \blacktriangleleft_{Tr,\sigma}$. Therefore, $\langle \odot, \odot\rangle \in \blacktriangleleft_{Tr,\sigma}$ will be satisfied only in the following case.

The substitution σ contains such substitution components $x_1 \mapsto t_1, \ldots, x_n \mapsto t_n$ ($n \geq 1$) that for parameters $y_1, \ldots, y_n$ from $t_1, \ldots, t_n$ respectively the following takes place: $x_1 \lessdot_{Tr} \odot$, $y_1 \lll_{Tr,\sigma} x_1$, $x_2 \prec_F y_1$, $y_2 \lll_{Tr,\sigma} x_2$, $x_3 \prec_F y_2$, $\ldots$, $y_n \lll_{Tr,\sigma} x_n$, and $\odot \lessdot_{Tr} y_n$.

This implies $x_1 \lessdot_{Tr} y_n$, i.e. $x_1 \lhd_{Tr,\sigma} y_n$ holds. Using the just given inequalities, we obtain that $x_1 \lhd_{Tr,\sigma} x_1$ by the transitivity of $\lhd_{Tr,\sigma}$. This contradicts the assumption on the irreflexivity of $\lhd_{Tr,\sigma}$. □

7. KANGER-TYPE CALCULI WITH ADMISSIBILITY AND COMPATIBILITY

The next step in our considerations is to improve the efficiency of quantifier handling in **KC+Mod**$_m$ using the above-given notions of admissibility and compatibility and rejecting from the applications of any quantifier rules. For this, we determine a family of special calculi, which, in opposite to $\mathbf{K}^{\approx}$ from [17] exploiting dummies and parameters as well as so-called substitution lists for determining Kanger's admissibility, uses the above-introduced notions of admissibility and computability.

This family is denoted by **BC+Mod**$_m$, where **BC** is determined in Fig. 1. Any its calculus contains all the rules of **KC+Mod**$_m$ except $(\forall \rightarrow)$ and $(\rightarrow \exists)$ that are replaced by $(\forall^{\#} \rightarrow)$ and $(\rightarrow \exists^{\#})$. We see that similar to **KC+Mod**$_m$, its special cases (with or without equality and modal rules) lead to the following calculi: $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$, $\mathbf{BK}+\mathbf{Mod}_m$, $\mathbf{BK}^{\approx}$, and **BK** for the classical case and $\mathbf{BJ}^{\approx}+\mathbf{Mod}_m$, $\mathbf{BJ}+\mathbf{Mod}_m$, $\mathbf{BJ}^{\approx}$, and **BJ** for the intuitionistic case, where **BK** and **BJ** denote classical and intuitionistic calculi respectively.

As in the case of **KC+Mod**$_m$, proof search in **BC+Mod**$_m$ is of the form of an inference tree for a starting sequent of the form $\rightarrow {}^1[_\nu F]$ and is performed in the Kanger style, i.e. it consists of two stages: *equality-independent* and *equality-dependent* ones.

The *equality-independent* stage includes two steps that have been described for **KC+Mod**$_m$, but in opposite to **KC+Mod**$_m$, it is completed by the construction of an inference tree, say, Tr, and the selection of a certain substitution, say, σ, replacing all the free dummies from Tr by terms without dummies. If not all the leaves of $Tr \cdot \sigma$ are axioms, then in the *equality-dependent* stage, only equality rules are applied to the non-axiom leaves of $Tr \cdot \sigma$, after this to their "heirs", and so on, attempting to construct a proof tree (i.e. a tree containing only axioms in leaves). In the case of success, the deduction is completed and Tr is declared as a *latent proof tree* for $\rightarrow {}^1[_\nu F]$ w.r.t. σ in $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$. Otherwise, the same is being repeated for $\rightarrow {}^1[_\nu F]$ again, if necessary.

The Kanger style proof search in **BC+Mod**$_m$ gives the possibility to impose some restrictions on a substitution that can be selected before going to the equality-dependent stage.

LEMMA 2. *Let Tr be a latent proof tree for a starting sequent $\rightarrow {}^1[_\nu F]$ w.r.t. a substitution σ in* **BC+Mod**$_m$. *Then σ can be considered as a substitution replacing all the free dummies in Tr by terms without dummies. Moreover, for any substitution component of $x \mapsto t \in \sigma$, x can be considered as to be a free dummy and t as to be a term containing only functional and constant symbols from $\rightarrow {}^1[_\nu F]$ (including, possibly, c_0) as well as free parameters from Tr.*

PROOF. Since a substitution component ${}^i_n x \mapsto c_0$, where ${}^i_n x$ is a free dummy not being a variable of σ and occurring in $Tr \cdot \sigma$ and, therefore, in Tr, can be added to σ, then σ can be considered as to replace all the free dummies in Tr and only them by terms without dummies.

Now suppose that in terms of σ there are functional symbols (including constant symbols distinct from c_0) not occurring in $\rightarrow {}^1[_\nu F]$. Then in Tr can be leaves with axioms containing such functional symbols and/or constants. These axioms are of the form $\underline{\Gamma, A \rightarrow A, \Delta}$, where A can supposed an atomic formula. All the rules of our calculi possess the subformula property. Therefore, in $\rightarrow {}^1[F]$ there are two atomic formulas, for copies A_1 and A_2 of which $A_1 \cdot \sigma = A_2 \cdot \sigma = A$. Hence, σ is a simultaneous unifier of all such pairs of atomic formulas.

Using the most general unifier algorithm from [40], it is possible to generate the most general simultaneous unifier σ' of all the just described pairs of atomic formulas with the following property satisfied: all the terms of σ' contain only functional and constant symbols (including, maybe, c_0) from $\rightarrow {}^1[_\nu F]$ as well as dummies and parameters from Tr and there is such a substitution λ that $\sigma = \sigma' \cdot \lambda$. This means that only terms of λ (not containing dummies) can contain functional and constant symbols not occurring in $\rightarrow {}^1[_\nu F]$.

Transform λ in a substitution λ' by means of the replacement of all its (maximal) terms beginning with such a functional or constant symbol by the constant c_0. For λ' we have that the terms of λ' do not contain dummies and $\sigma' \cdot \lambda'$ is a simultaneous unificator of all the above-mentioned pairs of atomic formulas.

Let us take in Tr such a leaf with a sequent S that $S \cdot \sigma$ is not an axiom. Then on the equality-dependent stage, the sequent $S \cdot \sigma (= (S \cdot \sigma') \cdot \lambda = S \cdot (\sigma' \cdot \lambda))$ can be transformed into an axiom by applying only equality rules. But in this case it is obvious that there exists a sequence of equality rules applications transforming the leaf $S \cdot (\sigma' \cdot \lambda')$ (without dummies) into an axiom of $\mathbf{BC{+}Mod}_m$. Since this takes place for any such S, for completing the proof, it remains to note that $\sigma' \cdot \lambda'$ can be taken as σ, to which is referred in the lemma. □

In what follows, the restriction on σ given in Lemma 2 is always supposed to be satisfied.

Proposition 2. *For a closed formula F, the initial sequent $\rightarrow F$ is deducible in the $\mathbf{KC{+}Mod}_m$ calculus if, and only if, an inference tree Tr for the starting sequent $\rightarrow {}^1[_\nu F]$ can be constructed at the equality-independent stage in the $\mathbf{BC{+}Mod}_m$ calculus and a substitution σ of terms without dummies for all the free dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent proof tree for $\rightarrow {}^1[_\nu F]$ w.r.t. σ in $\mathbf{BC{+}Mod}_m$, (ii) σ is admissible for Tr, (iii) Tr is compatible with σ w.r.t. $\mathbf{BC{+}Mod}_m$.*

Proof. First of all note that because of the construction of $\mathbf{KC{+}\ Mod}_m$ and $\mathbf{BC{+}Mod}_m$ (distinguished by the quantifier rules only), it is sufficient to prove Proposition 2 for the calculi $\mathbf{KK}^{\approx}$ and $\mathbf{BK}^{\approx}$. Moreover, because of the requirement of making proof search in both $\mathbf{KK}^{\approx}$ and $\mathbf{BK}^{\approx}$ in the Kanger style, it is sufficient to prove the following:

(*) If Tr is an inference tree for $\rightarrow {}^1[_\nu F]$ in $\mathbf{BK}$ and a substitution σ is selected in such a way that (ii) and (iii) are satisfied, then one can construct such an inference tree Tr' for $\rightarrow {}^1[_\nu F]$ in $\mathbf{KK}$ that $Tr \cdot \sigma$ coincides with Tr', and vice versa, if Tr' is an inference tree in $\mathbf{KK}$, then one can construct an inference tree Tr for $\rightarrow {}^1[_\nu F]$

in **BK** and select a substitution σ in such a way that (ii) and (iii) are satisfied and Tr' coincides with $Tr \cdot \sigma$.

Additionally note that Lemma 1 allows us to replace (ii) (requiring the irreflexivity of $\lhd_{Tr,\sigma}$) by the requirement of the irreflexivity of $\blacktriangleleft_{Tr,\sigma}$.

The following reasonings lead to proving (*) with the modified item (ii).

(=>) Remind that according to Lemma 2, σ replaces all the free dummies in Tr by terms without dummies and any such term can contain constants (including c_0), functional symbols, and free parameters from Tr. The formula F is closed. Therefore, any dummy of σ is introduced by the application of $(\forall^{\#} \to)$ or $(\to \exists^{\#})$. That is why it is sometimes convenient to write its substitution component as ${}^{(l+1)+i}_{\ \ \ \ n}\underline{x} \mapsto t_i$ (see the definitions of $(\forall^{\#} \to)$ and $(\to \exists^{\#})$ in Fig. 1).

By condition, the relation $\blacktriangleleft_{Tr,\sigma}$ is irreflective. Thus, $\blacktriangleleft_{Tr,\sigma}$ can be extended to a *strict linear order relation* $\blacktriangleleft^{+}_{Tr,\sigma}$ such that $\blacktriangleleft_{Tr,\sigma} \subseteq \blacktriangleleft^{+}_{Tr,\sigma}$. That is $\blacktriangleleft^{+}_{Tr,\sigma}$ takes place for any pair of distinct elements $\odot$ and $\odot'$ from $\blacktriangleleft_{Tr,\sigma}$ and $\langle \odot, \odot' \rangle \in \blacktriangleleft_{Tr,\sigma}$ implies $\langle \odot, \odot' \rangle \in \blacktriangleleft^{+}_{Tr,\sigma}$.

As a result, we obtain that $\blacktriangleleft^{+}_{Tr,\sigma}$ can be taken as a proper sequence for Tr providing the compatibility of Tr with σ w.r.t **BK**. This along with the linearity of $\blacktriangleleft^{+}_{Tr,\sigma}$ implies that if ${}^{(l+1)+i}_{\ \ \ \ n}\underline{x} \mapsto t_i \in \sigma$, then for any parameter ${}^{q}_{m}\overline{y}$ from t_i, $\langle {}^{q}_{m}\overline{y},\ {}^{(l+1)+i}_{\ \ \ \ n}\underline{x} \mapsto t_i \rangle \in \blacktriangleleft^{+}_{Tr,\sigma}$.

Keeping in mind the just said and Lemma 2, we conclude that the tree $Tr \cdot \sigma$ can be constructed by the "multiplication" of Tr by any arbitrarily selected substitution component of σ at the first step, the "multiplication" of the obtained result by any other arbitrarily selected substitution component of σ at the second step, and so on until all the substitution components of σ have been exhausted.

By this property of $Tr \cdot \sigma$, now it is easy to see that $Tr \cdot \sigma$ can be considered as an inference tree in the **KK** calculus, the way of the construction of which is determined by the rules applications order $\blacktriangleleft^{+}_{Tr,\sigma}$, which means that the first connective of $\blacktriangleleft^{+}_{Tr,\sigma}$ determines the selection of the first rule for its application for the construction of $Tr \cdot \sigma$, the second connective of $\blacktriangleleft^{+}_{Tr,\sigma}$ determines the selection of the second rule application for the construction of $Tr \cdot \sigma$, and so on until all the connectives of $\blacktriangleleft^{+}_{Tr,\sigma}$ have been looked through.

Indeed, the application of any rule of **BK** eliminating a connective in Tr and distinguished from both $(\forall^{\#} \to)$ and $(\to \exists^{\#})$ can be considered as the application of the same rule in $Tr \cdot \sigma$, but already in the **KK** calculus.

Let us consider the case of the rule $(\forall^{\#} \to)$ ($(\to \exists^{\#})$) that is applied at a node N of Tr after applying all other rules proceeding to this application according to the order $\blacktriangleleft^{+}_{Tr,\sigma}$.

Let it be of the form

$$\frac{\Gamma, {}^{i}_{n}\forall^{i}_{n}xA \to \Delta}{\Gamma, A|^{{}^{i}_{n}x}_{{}^{(l+1)+i}_{\ \ \ \ n}\underline{x}} \to \Delta}\ (\forall^{*} \to) \quad \Big(\frac{\Gamma \to {}^{i}_{n}\exists^{i}_{n}xA, \Delta}{\Gamma \to A|^{{}^{i}_{n}x}_{{}^{(l+1)+i}_{\ \ \ \ n}\underline{x}}, \Delta}\ (\to \exists^{*})\Big),$$

where for a dummy ${}^{(l+1)+i}_{\ \ \ \ n}\underline{x}$ there is a term t_i such that ${}^{(l+1)+i}_{\ \ \ \ n}\underline{x} \mapsto t_i \in \sigma$; moreover, as it was said earlier, for any parameter ${}^{q}_{m}\overline{y}$ from t_i, $\langle {}^{q}_{m}\overline{y},\ {}^{(l+1)+i}_{\ \ \ \ n}\underline{x} \rangle \in \blacktriangleleft^{+}_{Tr,\sigma}$, which means that the term t_i is free for ${}^{(l+1)+i}_{\ \ \ \ n}\underline{x}$ in $A|^{{}^{i}_{n}x}_{{}^{(l+1)+i}_{\ \ \ \ n}\underline{x}}$ and, therefore, t_i is free for ${}^{i}_{n}x$ in A. Hence, it can considered that at the node N of the tree $Tr \cdot \sigma$, the rule

$$\frac{\Gamma, {}^i_n\forall^i_n x A \to \Delta}{\Gamma, A|^{{}^i_n x}_{t_i} \to \Delta}\ (\forall \to) \quad (\frac{\Gamma \to {}^i_n\exists^i_n x A, \Delta}{\Gamma \to A|^{{}^i_n x}_{t_i}, \Delta}\ (\to \exists))$$

of **KK** satisfying the Gentzen admissibility is applied.

Thus, $Tr \cdot \sigma$ can be constructed according to the order $\blacktriangleleft^+_{Tr,\sigma}$. For completing the proving of the necessity in (*), it remains to take $Tr \cdot \sigma$ as Tr'.

(<=) Suppose that Tr' is an inference tree for $\to {}^1[_\nu F]$ in **KK** and $\alpha'_1, \ldots, \alpha'_k$ the sequence of rules applications in Tr' (among of which the applications of both $Con \to$ and $\to Con$ may occur) leading to the construction of Tr' when looking through it from left to right.

This means that the process of the construction of Tr' in **KK** can be considered as the subsequent construction of trees $Tr'_0, Tr'_1, \ldots, Tr'_k$, where Tr'_0 is the tree with the only root N_0 labeled by $\to {}^1[_\nu F]$, Tr'_1 is generated by the application of α'_1 to the unique leaf N_0 of Tr'_0, Tr'_2 is generated by the application of α'_2 to a leaf N_1 of Tr'_1, …, and Tr'_k (coinciding with Tr') is generated by the application of α'_k to a leaf N_{k-1} of Tr'_{k-1}.

Let ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ is a sequence of all the connectives being eliminated in Tr' by some applications from $\alpha'_1, \ldots, \alpha'_r$ and written from left to right according to their elimination ($r \le k$).

Let us make the subsequent construction of inference trees $Tr_0, Tr_1, \ldots, Tr_k$ in **BK**, simultaneously determining inference rules $\alpha_1, \ldots, \alpha_k$ of **BK** and defining substitutions $\sigma_0, \sigma_1, \ldots, \sigma_r$ in the following way.

(I) Tr_0 coincides with Tr'_0; $\sigma_0 = \emptyset$.

(II) Let Tr'_{j-1} be already constructed and σ_{j-1} be already defined ($1 \le j \le r$).

(II.1) Suppose α'_j is the application of $(\forall \to)$ ($(\to \exists)$) to the node N_{j-1} in Tr'_{j-1} and is of the form

$$\frac{\Gamma, {}^i_n\forall^i_n x A \to \Delta}{\Gamma, A|^{{}^i_n x}_{t_i} \to \Delta}\ (\forall \to) \quad (\frac{\Gamma \to {}^i_n\exists^i_n x A, \Delta}{\Gamma \to A|^{{}^i_n x}_{t_i}, \Delta}\ (\to \exists)),$$

where t_i is a term being a free for ${}^i_n x$ in A. Then α_j denotes the application of the following rule

$$\frac{\Gamma, {}^i_n\forall^i_n x A \to \Delta}{\Gamma, A|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}} \to \Delta}\ (\forall^* \to) \quad (\frac{\Gamma \to {}^i_n\exists^i_n x A, \Delta}{\Gamma \to A|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}}, \Delta}\ (\to \exists^*)),$$

to N_{j-1} in Tr_{j-1} , thereby generating Tr_j; σ_j is defined as the set $\sigma_{j-1} \cup \{{}^{(l+1)+}{}^i_n\underline{x} \mapsto t_i\}$ being a substitution because $l+1$ is a new index (w.r.t. Tr_{j-1}) and any two nodes of Tr_{j-1} lying in different branches of Tr_{j-1} have no common bound variables.

(II.2) In all the other cases, α_j denotes the application α'_j (i.e. the application of the same rule that was applied to N_{j-1} of Tr'_{j-1} for generating Tr'_j), Tr_j denotes the result of application α_j to Tr_{j-1} , and σ_j is defined as equal to σ_{j-1}.

If we take Tr_k as Tr and σ_k as σ, it is obvious that Tr is an inference tree in **BK**, σ is a substitution of terms without dummies for all the free dummies from Tr, and Tr' coincides with $Tr \cdot \sigma$.

Now we can prove that (ii) and (iii) from (*) take place.

Let us consider the sequence $\alpha_1, \ldots, \alpha_k$. According to its definition, each α_k is $(Con \rightarrow)$ or $(\rightarrow Con)$ or eliminates one of the connectives ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$; at that, these connectives and only they occur in Tr.

Obviously, ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ can be considered as a strict linear order (say, Λ) corresponding to the elimination of the connectives in Tr according to the subsequent applications of $\alpha_1, \ldots, \alpha_k$ in Tr. This implies that ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ is a proper sequence for Tr.

Suppose ${}^{(l+1)+i}{}_{n}^{}\underline{x} \mapsto t_i \in \sigma$ is a substitution component defined at the step (II.1). Since the term t_i is free for ${}^{i}_{n}x$ in A, then for any parameter ${}^{i}y$ from t_i, $\langle {}^{i}\overline{y}, {}^{i}_{n}x_{(l+1)+{}^{i}_{n}\underline{x}} \rangle \in \Lambda$, which proves the inclusion $\ll_{Tr,\sigma} \subseteq \Lambda$.

In its turn, if $\langle y, x\rangle \in \lessdot_{Tr}$, then y and x occur in Λ and $\langle y, x\rangle \in \Lambda$. This and the coincidence of the sets of all dummies and parameters from Λ and $\lessdot_{Tr}$ lead to the inclusion $\blacktriangleleft_{Tr,\sigma} \subseteq \Lambda$. Λ is a strict linear order. Therefore, it is irreflexive and it coincides with its transitive closer. Hence, $\blacktriangleleft_{Tr,\sigma} \subseteq \Lambda$, which means that $\blacktriangleleft_{Tr,\sigma}$ is an irreflexive relation. The proof of (ii) form (*) is completed.

For proving (iii) from (*) , it is enough to note that the inclusion $\blacktriangleleft_{Tr,\sigma} \subseteq \Lambda$ provides the compatibility of Tr with σ w.r.t. **BK**, since Tr was constructed according to the subsequent applications $\alpha_1, \ldots, \alpha_k$ containing the applications eliminating ${}_{j_1}\odot_1, \ldots, {}_{j_r}\odot_r$ in the accordance with the order Λ. $\square$

Now we can transform this proposition into its special forms.

Suppose F is a formula and σ a substitution. Let $\blacktriangleleft^-_{F,\sigma}$ denote all the pairs $\langle \odot, \odot' \rangle$ from $\blacktriangleleft_{F,\sigma}$ such that both $\odot$ and $\odot'$ are not quantifier connectives (or their variables). (The same concerns $\blacktriangleleft^-_{S,\sigma}$, $\blacktriangleleft^-_{\Xi,\sigma}$, and $\blacktriangleleft^-_{Tr,\sigma}$, where S is a sequent, Ξ a set of formulas or sequents, and Tr an inference tree.)

An inference tree Tr is called *propositionally compatible* with a substitution σ w.r.t. a sequent calculus **SC** if, and only if, the condition of the compatibility of Tr with σ w.r.t. **SC** is satisfied for any pair $\langle {}_{j_n}\odot_n, {}_{j_m}\odot_m \rangle$ from $\blacktriangleleft^-_{Tr,\sigma}$.

Theorem 1. *For a closed formula F, the initial sequent $\rightarrow F$ is deducible in an intuitionistic (modal) calculus* $\mathbf{KJ}^{\approx}$+$\mathbf{Mod}_m$ *if, and only if, an inference tree Tr for the starting sequent $\rightarrow {}^{1}[{}_{\nu}F]$ can be constructed at the equality-independent stage in* $\mathbf{BJ}^{\approx}$+$\mathbf{Mod}_m$ *and a substitution σ of terms without dummies for all the free dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent proof tree for $\rightarrow^1 [{}_{\nu}F]$ w.r.t. σ in* $\mathbf{BJ}^{\approx}$+$\mathbf{Mod}_m$*, (ii) σ is admissible for Tr, (iii) Tr is propositionally compatible with σ w.r.t.* **BJ**+$\mathbf{Mod}_m$.

Proof. According to Proposition 2, this theorem will take place if we show that the compatibility from (iii) of Proposition 2 can be replaced by the propositional compatibility w.r.t. **BJ**+$\mathbf{Mod}_m$.

On the basis of Lemma 1, the admissibility of σ for Tr is equivalent to the irreflexivity of $\blacktriangleleft_{Tr,\sigma}$. This means that in the case of the admissibility of the substitution σ for the tree Tr in **BJ** + $\mathbf{Mod}_m$, it is enough to check the propositional compatibility of the Tr with σ. (The restriction providing the intuitionistic case in the paper consists in that the succedent of any deduced sequent should contain no more than one formula.)

Since proof search in $\mathbf{BJ}^{\approx}$+$\mathbf{Mod}_m$ is always made in the Kanger style, the applications of the equality rules (and only them) on the equality-dependent stage have no influence on the relation $\blacktriangleleft_{Tr,\sigma}$ generated earlier. This means that it is enough to check the compatibility of the tree Tr with the substitution σ only w.r.t. $\mathbf{BJ}+\mathbf{Mod}_m$. $\Box$

As for the classical case, when there are no restrictions on on the form of sequents and orders of propositional and modal rule applications, Proposition 2 takes the following form.

THEOREM 2. *For a closed formula F, the initial sequent $\to F$ is deducible in a classical (modal) calculus $\mathbf{KK}^{\approx}+\mathbf{Mod}_m$ if, and only if, an inference tree Tr for the starting sequent $\to {}^{1}[_{\nu}F]$ can be constructed at the equality-independent stage in $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$ and a substitution σ of terms without dummies for all the free dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent proof tree for $\to^{1} [_{\nu}F]$ w.r.t. σ in $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$ and (ii) σ is admissible for Tr.*

PROOF. Since there are no restrictions on the form of sequents and orders of applications of logical and modal rules in $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$, the item (iii) in Proposition 2 becomes redundant: any latent proof tree Tr constructed in $\mathbf{BK}^{\approx}+\mathbf{Mod}_m$ is compatible with any σ if, and only if, σ is an admissible for Tr. $\Box$

EXAMPLE 1. Let us demonstrate the peculiarities of proof search in $\mathbf{BC}+\mathbf{Mod}_m$ for its concretizations as the calculi $\mathbf{BK}^{\approx}$ and $\mathbf{BJ}^{\approx}$.

Consider the classical-type modal calculus **GK** and intuitionistic-type modal calculus **JGK** from [39]. In our notation, **GK** with equality can be determined as $\mathbf{KK}^{\approx}+\{(\Box)\}$ and **JGK** with equality as $\mathbf{KJ}^{\approx}+\{(\Box),(\Diamond)\}$, where $(\Box)$ is the modal rule $\dfrac{\Box\Gamma \to \Box G}{\Gamma \to G}$ and $(\Diamond)$ the modal rule $\dfrac{\Diamond\Gamma \to \Diamond G}{\Gamma \to G}$. (Here, G is a formula, Γ is a multiset of formulas $F_1, \ldots, F_k$, and $\Box\Gamma$ and $\Diamond\Gamma$ denote multisets of formulas $\Box F_1, \ldots, \Box F_k$ and $\Diamond F_1, \ldots, \Diamond F_k$ respectively.)

Consider the formula $H^* = \exists y \neg \Box \exists x (y \approx f(y) \supset P(x, y)) \supset \neg \forall y' \Box \exists x' P(x', f(y'))$ being the modification of F^* for the case of logics with equality.

Looking along H^* from left to right, we are "numbering" all its connectives with the help of left down indexes in the following way: $\exists y$ receives 1, $\neg$ receives 2, ..., $\exists x'$ receives 10.

Now we can construct the following tree Tr for $\to {}^{1}[H^*]$ being an inference tree in both $\mathbf{BK}^{\approx} + \{(\Box)\}$ and $\mathbf{BJ}^{\approx} + \{(\Box),(\Diamond)\}$ (here, ${}^{1}[H^*] = {}^{1}_{1}\exists^{1}_{1}y{}^{1}_{2}\neg{}^{1}_{3}\Box^{1}_{4}\exists^{1}_{4}x({}^{1}_{1}y \approx f({}^{1}_{1}y)\,{}^{1}_{5}\supset P({}^{1}_{4}x,{}_{1}^{1}y))\,{}^{1}_{6}\supset {}^{1}_{7}\neg{}^{1}_{8}\forall^{1}_{8}y'{}^{1}_{9}\Box^{1}_{10}\exists^{1}_{10}x'P({}^{1}_{10}x', f({}^{1}_{8}y')))$:

$$\to {}^{1}[H^*] \qquad \text{(a starting sequent)}$$

$${}^{1}_{1}\exists^{1}_{1}y{}^{1}_{2}\neg{}^{1}_{3}\Box^{1}_{4}\exists^{1}_{4}x({}^{1}_{1}y \approx f({}^{1}_{1}y)\,{}^{1}_{5}\supset P({}^{1}_{4}x,{}_{1}^{1}y)) \to {}^{1}_{7}\neg{}^{1}_{8}\forall^{1}_{8}y'{}^{1}_{9}\Box^{1}_{10}\exists^{1}_{10}x'P({}^{1}_{10}x', f({}^{1}_{8}y')) \qquad \text{(by } (\to\supset))$$

$${}^{3}_{2}\neg{}^{3}_{3}\Box^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}y \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})) \to {}^{1}_{7}\neg{}^{1}_{8}\forall^{1}_{8}y'{}^{1}_{9}\Box^{1}_{10}\exists^{1}_{10}x'P({}^{1}_{10}x', f({}^{1}_{8}y')) \qquad \text{(by } (\exists\to))$$

$${}^{3}_{2}\neg{}^{3}_{3}\Box^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}y \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})),\ {}^{1}_{8}\forall^{1}_{8}y'{}^{1}_{9}\Box^{1}_{10}\exists^{1}_{10}x'P({}^{1}_{10}x', f({}^{1}_{8}y')) \to \qquad \text{(by } (\to\neg))$$

$${}^{3}\neg{}^{3}_{3}\Box^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}y \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})),\ {}^{5}_{9}\Box^{5}_{10}\exists^{5}_{10}x'P({}^{5}_{10}x', f({}^{5}_{8}\underline{y}')) \to \qquad \text{(by } (\forall\to))$$

$${}^{5}_{9}\Box^{5}_{10}\exists^{5}_{10}x'P({}^{5}_{10}x', f({}^{5}_{8}\underline{y}')) \to {}^{3}_{3}\Box^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})) \qquad \text{(by } (\neg\to))$$

$${}^{5}_{10}\exists^{5}_{10}x'P({}^{5}_{10}x', f({}^{5}_{8}\underline{y}')) \to {}^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})) \qquad \text{(by } (\Box))$$

$$P({}^{11}_{10}\overline{x'}, f({}^{5}_{8}\underline{y}')) \to {}^{3}_{4}\exists^{3}_{4}x({}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y})\,{}^{3}_{5}\supset P({}^{3}_{4}x,{}^{3}_{1}\overline{y})) \qquad \text{(by } (\to\exists))$$

$$P({}^{11}_{10}\overline{x'}, f({}^{5}_{8}\underline{y}')) \to ({}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y}))\,{}^{15}_{5}\supset P({}^{15}_{4}\underline{x},{}^{3}_{1}\overline{y}) \qquad \text{(by } (\exists\to))$$

$$P({}^{11}_{10}\overline{x'}, f({}^{5}_{8}\underline{y}')), {}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y}) \ \rightarrow P({}^{15}_{4}\underline{x}, {}^{3}_{1}\overline{y})) \qquad \text{(by } (\rightarrow\supset)\text{)}$$

According to definitions, x and y' are dummies in H^*, while x' and y parameters (in H^*). Hence, ${}^{1}_{4}x$, ${}^{3}_{4}x$, ${}^{15}_{4}x$, ${}^{1}_{8}y'$, and ${}^{5}_{8}y'$ are dummies while ${}^{1}_{10}x'$, ${}^{5}_{10}x'$, ${}^{11}_{10}x'$, ${}^{1}_{1}y$, and ${}^{3}_{1}y$ are parameters.

The application of the substitution $\sigma = \{{}^{15}_{4}\underline{x} \mapsto {}^{11}_{10}\overline{x'},\ {}^{5}_{8}\underline{y'} \mapsto {}^{3}_{1}\overline{y}\}$ to Tr transforms the unique leaf of Tr into $P({}^{11}_{10}\overline{x'},\ f({}^{3}_{1}\overline{y})),\ {}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y}) \rightarrow P({}^{11}_{10}\overline{x'}, {}^{3}_{1}\overline{y})$. By applying the rule $(\rightarrow\approx)$ to this sequent w.r.t. ${}^{3}_{1}y \approx f({}^{3}_{1}y)$, we can deduce the axiom $P({}^{11}_{10}\overline{x'}$, ${}^{3}_{1}\overline{y}),\ {}^{3}_{1}\overline{y} \approx f({}^{3}_{1}\overline{y}) \rightarrow P({}^{11}_{10}\overline{x'}, {}^{3}_{1}\overline{y})$. Thus, Tr is a latent proof tree w.r.t. σ in both $\mathbf{BK}^{\approx} + \{(\Box)\}$ and $\mathbf{BJ}^{\approx} + \{(\Box), (\Diamond)\}$. We can check the admissibility of σ for Tr and propositional compatibility of Tr with σ w.r.t. $\mathbf{BJ}^{\approx} + \{(\Box), (\Diamond)\}$. By Theorems 1 and 2, we obtain the deducibility of $\rightarrow H^*$ in both $\mathbf{KK}^{\approx} + \{(\Box)\}$ and $\mathbf{KJ}^{\approx} + \{(\Box), (\Diamond)\}$.

For the formula $H_1^* = \neg\forall y \Box \exists x (y \approx f(y) \supset P(x, y)) \supset \exists y' \neg \Box \exists x'\ P(x',\ f(y'))$, being a modification of H^*, Tr can be easily transformed into a latent proof tree Tr' for $\rightarrow {}^{1}[H_1^*]$ w.r.t. σ' in both $\mathbf{BK}^{\approx} + \{(\Box)\}$ and $\mathbf{BJ}^{\approx} + \{(\Box), (\Diamond)\}$, where σ' is the corresponding transformation of σ. The substitution σ' will be admissible for Tr'. Therefore, $\rightarrow H_1^*$ becomes deducible in $\mathbf{KK}^{\approx} + \{(\Box)\}$ by Theorem 2. At the same time, it can be proved that for $\rightarrow {}^{1}[H_1^*]$, neither Tr' nor any other latent proof tree will be propositionally compatible with σ' or any other "reasonable" substitution. Hence, by Theorem 1, $\rightarrow H_1^*$ is not deducible in $\mathbf{KJ}^{\approx} + \{(\Box), (\Diamond)\}$.

For the formula $H_2^* = \forall y \neg \forall x \Box (y \approx f(y) \supset P(x, y)) \supset \neg \forall y' \Box \exists x'\ P(x',\ f(y'))$ being another modification of H^*, Tr can be transformed into Tr'' being a latent tree for $\rightarrow {}^{1}[H_2^*]$ w.r.t. corresponding σ'' in both the calculi. At that, σ'' will not be admissible for Tr''. Moreover, it can be proved that for any other latent proof tree for $\rightarrow {}^{1}[H_2^*]$, there is no substitution admissible for it. Hence, by Theorems 1 and 2, $\rightarrow H_2^*$ cannot be deduced neither in $\mathbf{BK}^{\approx} + \{(\Box)\}$ nor in $\mathbf{BJ}^{\approx} + \{(\Box), (\Diamond)\}$.

Pay your attention to the fact that the order of quantifier rules applications in Tr from Example 1 is immaterial; it can be any. The other peculiarity is that the quantifier rules simply introduce new parameters and dummies participating in checking admissibility and compatibility. This observation serves as a "guide" to constructing quantifier-rule-free sequent calculi.

8. QUANTIFIER-RULE-FREE SEQUENT CALCULI

Now we can move to the construction of first-order quantifier-rule-free sequent calculi on the basis of the introduced notions of admissibility and compatibility.

At first sight it may seem that the best way to achieve this is to simply initiate a search of a sequent $\rightarrow \mu({}^{1}[_{\nu}F])$ in $\mathbf{BC}$+$\mathbf{Mod}_m$ without using its quantifier rules. Obviously, with this approach, Proposition 2 holds for all the sequents deduced in $\mathbf{KC}$+$\mathbf{Mod}_m$. However, it also turns out to be valid for some sequences, which are not deducible in $\mathbf{KC}$+$\mathbf{Mod}_m$. This is coursed by that during the transformation of $\rightarrow F$ into $\rightarrow \mu({}^{1}[_{\nu}F])$, the loss of the information about locations of quantifiers in F is observed, while this information is used in proof search in $\mathbf{BC}$+$\mathbf{Mod}_m$ for renaming certain variables when applying this or that quantifier rule.

For example, for $F_1^* = ((P(a) \vee P(b)) \supset \exists x P(x))$ and $F_2^* = \exists x((P(a) \vee P(b)) \supset P(x))$, where a and b are constants and x a dummy, both the sequents $\rightarrow F_1^*$ and $\rightarrow F_2^*$ are deducible in $\mathbf{KK}$, while $\rightarrow F_1$ is deducible in $\mathbf{KJ}$ and $\rightarrow F_2$ is not. For $\mu({}^1[{}_\nu F_1^*]) = (P(a)\ {}^1_1\!\vee\ P(b))\ {}^1_2\!\supset\ P({}^1_3x)$ and $\mu({}^1[{}_\nu F_2^*]) = (P(a)\ {}^1_2\!\vee\ P(b))^1_3 \supset P({}^1_1x)$, both the sequents $\to \mu({}^1[{}_\nu F_1^*])$ and $\to \mu({}^1[{}_\nu F_2^*])$ are not deducible even in **BK**, despite ${}^1_1x$ and ${}^1_3x$ are dummies: after removing the quantifier rules, there is no possibility to produce the necessary copies of $P({}^1_1x)$ and $P({}^1_3x)$ with new dummies. On the other hand, if we allow doubling, for example, all atomic formulas with possible reindexing all their dummies and parameters, we can establish the deductibility of sequences $\to \mu({}^1[{}_\nu F_1^*])$ and $\to \mu({}^1[{}_\nu F_2^*])$ in both **BK** and **BJ** and, as a result, establish the deducibility of $\to F_1^*$ and $\to F_2^*$ in both **KK** and **KJ** on the basis of Theorems 1 and 2. But, as it was mentioned above, the sequent $\to F_2^*$ is not deducible in **KJ**. So, we should develop a special technique for indexing variables that takes into account the locations of the quantifiers in an original formula.

Let formulas G and F be taken from the definition of ${}^k_F[G]$, where k is a natural number ($k \geq 1$). Suppose there is such an ex-variable (i.e., parameter or dummy) ${}^i_nx$ for G relative to F that there is no external connectives $\odot$ (including the case of a variable) for G w.r.t. F, for which ${}^i_nx \lessdot_F \odot$ takes place. Then the *formula* ${}^{i}_{n}{}^{x<k}{}_F[G]$ is defined in the following way: it coincides with $a^k_F[G]$, in which all occurrences of ${}^i_nx$ are replaced simultaneously by a variable ${}^{k+i}_{\ \ n}x$. In all other cases, ${}^{i}_{n}{}^{x<k}{}_F[G]$ is declared to be equal to ${}^k_F[G]$.

For short we write ${}^{i}_{n}{}^{x<k}[G]$ instead of ${}^{i}_{n}{}^{x<k}_{F}[G]$ in the cases when F is considered to be known.

Analogously to ${}^k_F[\cdot]$, the operation ${}^{x<k}_{F}[\cdot]$ converts graphically distinct connectives and variables into graphically distinct connectives and variables, inducing a renaming of parameters with parameters only and dummies with dummies only.

For example, for the formula $G = \neg^1\Box^1\exists^1x\ P({}^1x, f({}^1y))$ (with the missed left down indexes) being a copy of the subformula $\neg\Box\exists xP(x, f(y))$ of F^*, the formulas ${}^2[G] = {}^2\neg^3\Box^3\exists^3xP({}^3x, f({}^1y))$ and ${}^{1}{}^{y<2}[G] = {}^2\neg^3\Box^3\exists^3xP({}^3x, f({}^3\overline{y}))$ (${}^2[G]$ and ${}^{1}{}^{y<2}[G]$ are distinguished by the variables ${}^1y$ and ${}^3\overline{y}$) only. If $G' = P({}^3x, f({}^1y))$, then ${}^2[G'] = P({}^3x, f({}^1y))$ and ${}^{3}{}^{x<2}[G'] = P({}^5\underline{x}, f({}^1y))$. If $G'' = {}^3\exists^3xP({}^3x, f({}^3y))$, then ${}^2[G''] = {}^5\exists^5xP({}^5x, f({}^3y) =^{z<2} [G'']$ for any variable z.

The operation ${}^{x<k}_{F}[\cdot]$ makes it possible to refuse all the quantifier rules, replacing them by the rules $(Con^{\#} \to)$ and $(\to Con^{\#})$ (see Fig. 1). As a result, we get a family of first-order quantifier-rule-free sequent calculi denoted by $\mathbf{QC{+}Mod}_m$, where **QC** contains all the propositional, contraction, and equality rules as well as all the axioms.

Analogously to $\mathbf{BC{+}Mod}_m$, we get the following modifications of $\mathbf{QC{+}Mod}_m$: $\mathbf{QK}^{\approx}\mathbf{{+}Mod}_m$, $\mathbf{QJ}^{\approx}\mathbf{{+}Mod}_m$, $\mathbf{QK{+}Mod}_m$, $\mathbf{QK{+}Mod}_m$, $\mathbf{QK}^{\approx}$, $\mathbf{QJ}^{\approx}$, **QK**, **QJ**, where **QK** and **QJ** denote the calculi for classical and intuitionistic logics without equality respectively.

Note that inference search in $\mathbf{QC{+}Mod}_m$ is made in the Kanger style, the equality-independent stage of which, due to the absence of quantifiers, consists of the second step only.

PROPOSITION 3. *For a closed formula F, the initial sequent $\to F$ is deducible in the* $\mathbf{KC{+}Mod}_m$ *calculus if, and only if, an inference tree Tr for $\to \mu({}^1[{}_\nu F])$ can be constructed at the equality-independent stage in the* $\mathbf{QC{+}Mod}_m$ *calculus and a substitution σ of terms without dummies for all the free dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent* *proof tree for* $\rightarrow \mu({}^1[{}_\nu F])$ *w.r.t.* σ *in* $\mathbf{QC{+}Mod}_m$*, (ii)* σ *is admissible for* Tr*, (iii)* Tr *is propositionally compatible with* σ *w.r.t.* $\mathbf{QC{+}Mod}_m$*.*

Proof. By Proposition 2, $\rightarrow {}^1[{}_\nu F]$ is deducible in $\mathbf{KC{+}Mod}_m$ if, and only if, an inference tree Tr' can be produced in $\mathbf{BC{+}Mod}_m$ and a substitution σ' of terms without dummies for all the free dummies from Tr can be selected in such a way that (i), (ii), and (iii) from Proposition 2 take place for Tr' and σ'. Obviously, it can be assumed that Tr' is constructed from top to bottom and left to right and that all the leaves of Tr' do not contain quantifiers and some part of these leaves are axioms, while the other leaves, not being axioms, can be transformed into axioms by applying equality rules at the equality-dependent stage.

Let us consider a sequent tree $\mu(Tr')$ being the result of the application of μ to all the sequents of Tr'. Denote it by Tr ($Tr = \mu(Tr')$). The sequent $\rightarrow \mu({}^1[{}_\nu F])$ lies in its root. Besides, it is quantifier-free and contains the same axioms as Tr' contains. At the equality-dependent stage, all leaves of Tr not being axioms are transformed into axioms. This means that for proving that Tr is a latent proof tree for $\rightarrow^1 [{}_\nu F]$ w.r.t. σ in $\mathbf{QC{+}Mod}_m$, it is enough to prove that any sequent of Tr not being its root is deduced by an inference rule.

Suppose a sequent S from Tr' is deduced by a propositional rule, modal rule, or contraction rule (either $(\rightarrow Con)$ or $(Con \rightarrow)$). Then, on the basis of the properties of μ, we conclude that the quantifier-free sequent $\mu(S)$ can be considered as it was deduced in $\mathbf{QC{+}Mod}_m$ by the application of the same propositional, modal, or contraction rule that was used when deducing S in $\mathbf{KC{+}Mod}_m$.

Suppose S is deduced in Tr' by one of the quantifier rules, say, for certainty, by $(\forall^\# \rightarrow)$: $\dfrac{\Gamma, {}^i_n\forall^i_n x A \rightarrow \Delta}{\Gamma, {}^{l+1}[A]|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}} \rightarrow \Delta}$. (That is $S = \Gamma, {}^{l+1}[A]|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}} \rightarrow \Delta$). Then Tr contains $\dfrac{\mu(\Gamma), \mu(A) \rightarrow \mu(\Delta)}{\mu(\Gamma), {}^{l+1}[\mu(A)]|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}} \rightarrow \mu(\Delta)}$, being the application of $(Con^\# \rightarrow)$, since ${}^{l+1}[\mu(A)]|^{{}^i_n x}_{(l+1)+{}^i_n\underline{x}}$ is another form of the formula ${}^i_n x^{<(l+1)}[\mu(A)]$.

The consideration of all the possible applications of the other quantifier rules in Tr' is made similar. This lead to that quantifier rules applications in Tr' can be replaced by the applications of an appropriate contraction rule $(Con^\# \rightarrow)$ or $(\rightarrow Con^\#)$ finally producing Tr.

Hence, Tr is a latent proof tree for $\rightarrow \mu({}^1[{}_\nu F])$ w.r.t. σ in $\mathbf{QC{+}Mod}_m$. According to the construction of Tr, we have that the sets of free variables of Tr and Tr' are coincided and that $\blacktriangleleft_{Tr,\sigma} = \blacktriangleleft_{Tr',\sigma}$. Thus, (i), (ii), and (iii) take place for Tr and σ in the case of $\mathbf{QC{+}Mod}_m$.

The poof in one direction is completed.

Now suppose that σ is a substitution and Tr an inference tree for $\rightarrow \mu({}^1[{}_\nu F])$ in $\mathbf{QC{+}Mod}_m$ satisfying (i), (ii), and (iii). Convert Tr into a sequent tree Tr' in the $\mathbf{BC{+}Mod}_m$ calculus in a way, similar to the just given above, but starting this conversation with the leaves of Tr and moving through Tr from right to left and bottom to top.

This means that all the leaves of Tr are declared as leaves of Tr' and any application of a propositiona,l, modal , or one of the contraction rules $(\rightarrow Con)$ and $(Con \rightarrow)$ in Tr is replaced by the application of the same inference rule, but already in the $\mathbf{BC{+}Mod}_m$ calculus. As to the contraction rules $(Con^{\#} \rightarrow)$ and $(\rightarrow Con^{\#})$, the application of any of them is replaced by the application of a certain quantifier rule from $\mathbf{BC{+}Mod}_m$, the selection of which is determined by which of the rules $(Con^{\#} \rightarrow)$ or $(\rightarrow Con^{\#})$ is under consideration and the type of a variable ${}^i_n x$ (dummy or parameter) introduced by this application.

Suppose for certainty that at some step of the construction of Tr', the rule $(\rightarrow Con^{\#})$ was applied (the consideration of the other cases of the possible applications of $(Con^{\#} \rightarrow)$ and $(\rightarrow Con^{\#})$ is similar): $\dfrac{\Gamma \rightarrow A, \Delta}{\Gamma \rightarrow {}^{l+1}[A]|^{{}^i_n x}_{(l+1)+{}^i_n \overline{x}}, \Delta}$ producing the sequent $\Gamma \rightarrow {}^{l+1}[A]|^{{}^i_n x}_{(l+1)+{}^i_n \overline{x}}, \Delta$, in which the formula ${}^{l+1}[A]|^{{}^i_n x}_{(l+1)+{}^i_n \overline{x}}$ is already generated in Tr' (remind that the construction of Tr' in made from right to left and bottom to top). Then this application of $(\rightarrow Con^{\#})$ in Tr becomes the $(\rightarrow \forall)$-rule application: $\dfrac{\Gamma \rightarrow {}^i_n\forall^i_n x A', \Delta}{\Gamma \rightarrow {}^{l+1}[A']|^{{}^i_n x}_{(l+1)+{}^i_n \overline{x}}, \Delta}$ in Tr'.

Making all such replacements of inference rule applications, when moving in Tr from right to left and bottom to top, we finally produce the root of Tr' with the sequent $\rightarrow {}^1[{}_\nu F]$.

According to the construction of Tr', Tr' is a latent proof tree for $\rightarrow {}^1[{}_\nu F]$ w.r.t. σ in $\mathbf{BC}^{\approx}\mathbf{{+}Mod}_m$. Besides, the sets of free variables of Tr and Tr' coincide and $\blacktriangleleft_{Tr,\sigma} = \blacktriangleleft_{Tr',\sigma}$, which means that (i), (ii), and (iii) take place for Tr' and σ in $\mathbf{BC{+}Mod}_m$. By Proposition 2, $\rightarrow F$ is deducible in the calculus $\mathbf{KC{+}Mod}_m$. □

Now we can reformulate Proposition 3 for the case of our intuitionistic modal logics in the following way, the proof of which "repeats" the proof of Theorems 1.

Theorem 3. *For a closed formula F, the initial sequent $\rightarrow F$ is deducible in the intuitionistic modal calculus $\mathbf{KJ}^{\approx}\mathbf{{+}Mod}_m$ if, and only if, an inference tree Tr for the sequent $\rightarrow \mu({}^1[{}_\nu F])$ can be constructed at the equality-independent stage in $\mathbf{QJ}^{\approx}\mathbf{{+}Mod}_m$ and a substitution σ of terms without dummies for all the dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent proof tree for $\rightarrow^1 [{}_\nu F]$ w.r.t. σ in $\mathbf{QJ}^{\approx}\mathbf{{+}Mod}_m$, (ii) σ is admissible for Tr, (iii) Tr is propositionally compatible with σ w.r.t. $\mathbf{QJ{+}Mod}_m$.*

As for the classical case, when there are no any restrictions on the form of sequents and order of propositional and modal rule applications, Proposition 3 takes the following form.

Theorem 4. *For a closed formula F, the initial sequent $\rightarrow F$ is deducible in the classical modal calculus $\mathbf{KK}^{\approx}\mathbf{{+}Mod}_m$ if, and only if, an inference tree Tr for the sequent $\rightarrow \mu({}^1[{}_\nu F])$ can be constructed at the equality-independent stage in $\mathbf{QK}^{\approx}\mathbf{{+}Mod}_m$ and a substitution σ of terms without dummies for all the dummies from Tr can be selected in such a way that the following conditions take place: (i) Tr is a latent proof tree for $\rightarrow^1 [{}_\nu F]$ w.r.t. σ in $\mathbf{QC}^{\approx}\mathbf{{+}Mod}_m$ and (ii) σ is admissible for Tr.*

The below-given examples give a possibility to gain some insight about proof search in our quantifier-rule-free calculi.

EXAMPLE 2. Let us demonstrate some peculiarities of the applications of $(\to Con^*)$ and $(Con^* \to)$ in $\mathbf{QC}^{\approx} + \mathbf{Mod}_m$ using F_1^* and F_2^* given at the beginning of this section. For them we have $\mu({}^1[_\nu F_1^*]) = ((P(a)\ {}^1_1\!\vee\ P(b))\ {}^1_2 \supset P({}^1_3x))$ and $\mu({}^1[_\nu F_2^*]) = ((P(a)\ {}^1_2\!\vee\ P(b))\ {}^1_3 \supset P({}^1_1x))$.

For $\to \mu({}^1[_\nu F_1^*])$, we can construct the below-given tree Tr being an inference tree Tr in both **QJ** and **QK**:

$$\dfrac{\dfrac{\dfrac{\to (P(a){}^1_1\!\vee P(b)){}^1_2 \supset P({}^1_3x)}{P(a){}^1_1\!\vee P(b) \to P({}^1_3x)}}{\dfrac{P(a) \to P({}^1_3x)}{P(a) \to P({}^3_3x), P({}^1_3x)} \qquad P(b) \to P({}^1_3x)}$$

(a starting sequent)
(by $(\to\supset)$)
(by $(\vee \to)$)
(by $(\to Con^*)$, when $n = 3, i = l = 1$, $F = F_1^*$, and $A = P({}^1_1x)$ (see definition))

The application of the substitution $\sigma = \{{}^1_3x \mapsto b, {}^3_3x \mapsto a\}$ converts the leaves of this tree into axioms. Hence, Tr is a latent proof tree. Obviously, σ is admissible for Tr; moreover, Tr is compatible with σ both w.r.t. **QJ**. Theorems 3 and 4 provide the deducibility of $\to F_1^*$ in **KJ** and **KK**. (As a result, we obtain the deducibility of $\to F_1^*$ in Gentzen's calculi **LJ** and **LK**.)

If for constructing a latent proof tree for $\to \mu({}^1[_\nu F_2^*])$ in **QJ** we will try to apply $(\to Con)$ or $(\to Con^*)$ to $\to (P(a)\ {}^1_2\!\vee P(b)){}^1_3 \supset P({}^1_1x)$, we will be able to deduce only copies of $\to \mu({}^1[_\nu F_2^*])$. Therefore, their applications will give nothing new for our purpose and it only remains to apply propositional rules. As a result, we will be able to construct the following inference tree:

$$\dfrac{\dfrac{\to (P(a){}^1_2\!\vee P(b)){}^1_3 \supset P({}^1_1x)}{P(a){}^1_2\!\vee P(b) \to P({}^1_1x)}}{P(a) \to P({}^1_1x) \qquad P(b) \to P({}^1_1x)}$$

(a starting sequent)
(by $(\to\supset)$)
(by $(\vee \to)$)

For converting two last sequents into axioms, it is necessary that ${}^1_1x \mapsto a$ and ${}^1_1x \mapsto b$ take place, which is impossible. Hence, $\to \mu({}^1[_\nu F_2^*])$ is not deducible in **QJ**. On the basis of Theorem 3, it cannot be deduced in **KJ** and, as a result, $\to F_2^*$ is not deducible in **LJ**.

The sequent $\to \mu({}^1[_\nu F_2^*])$ is deducible in **QK**, since $(\to Con^*)$ can be applied to the starting sequent $\to (P(a)\ {}^1_2\!\vee\ P(b))\ {}^1_3 \supset P({}^1_1x)$, giving the possibility to produce the following inference tree:

$$\dfrac{\dfrac{\dfrac{\dfrac{\to (P(a){}^1_2\!\vee P(b)){}^1_3 \supset P({}^1_1x)}{\to (P(a){}^1_2\!\vee P(b)){}^1_3 \supset P({}^1_1x), (P(a){}^3_2\!\vee P(b)){}^3_3 \supset P({}^3_1x)}}{P(a){}^1_2\!\vee P(b) \to P({}^1_1x), (P(a){}^3_2\!\vee P(b)){}^3_3 \supset P({}^3_1x)}}{P(a){}^1_2\!\vee P(b), P(a){}^3_2\!\vee P(b) \to P({}^1_1x), P({}^3_1x)}}{P(a), P(a){}^2_2\!\vee P(b) \to P({}^1_1x), P({}^3_1x) \qquad P(b), P(a){}^3_2\!\vee P(b) \to P({}^1_1x), P({}^3_1x)}$$

(a starting sequent)
(by $(\to Con^*)$)
(by $(\to\supset)$)
(by $(\to\supset)$)
(by $(\vee \to)$)

This tree is a latent proof tree, since the substitution $\{{}^1_1x \mapsto a, {}^3_1x \mapsto b\}$ converts two last sequents into axioms. Obviously, it is admissible for the constructed tree. Therefore, the sequent $\to\ {}^1[_\nu F_2^*]$ is deducible in the **KK** calculus by Theorem 4 and, as a result, the sequent $\to F_2^*$ is deducible in **LK**.

EXAMPLE 3. For the formula H^* from Example 1, we can construct the following inference tree Tr for $\to \mu(H^*)$ in **QC**.

$$\dfrac{\to\ {}^1_2\neg{}^1_3\Box({}^1_1\overline{y} \approx f({}^1_1\overline{y})\ {}^1_5 \supset P({}^1_4\underline{x}, {}^1_1\overline{y}))\ {}^1_6 \supset {}^1_7\neg{}^1_9\Box P({}^1_{10}\overline{x}', f({}^1_8\underline{y}'))}{}$$

(a starting sequent)

$$\frac{{}^{1}_{2}\neg^{1}_{3}\Box({}^{1}_{1}\overline{y} \approx f({}^{1}_{1}\overline{y})\ {}^{1}_{5} \supset P({}^{1}_{4}\underline{x}, {}^{1}_{1}\overline{y})) \rightarrow\ {}^{1}_{7}\neg^{1}_{9}\Box P({}^{1}_{10}\overline{x}', f({}^{1}_{8}\underline{y}'))}{{}^{1}_{2}\neg^{1}_{3}\Box({}^{1}_{1}\overline{y} \approx f({}^{1}_{1}\overline{y})\ {}^{1}_{5} \supset P({}^{1}_{4}\underline{x}, {}^{1}_{1}\overline{y})),\ {}^{1}_{9}\Box P({}^{1}_{10}\overline{x}', f({}^{1}_{8}\underline{y}')) \rightarrow} \qquad \text{(by } (\rightarrow\supset))$$

$$\frac{}{{}^{1}_{9}\Box P({}^{1}_{10}\overline{x}', f({}^{1}_{8}\underline{y}')) \rightarrow\ {}^{1}_{3}\Box({}^{1}_{1}\overline{y} \approx f({}^{1}_{1}\overline{y})\ {}^{1}_{5} \supset P({}^{1}_{4}\underline{x}, {}^{1}_{1}\overline{y}))} \qquad \text{(by } (\rightarrow \neg))$$

$$\frac{}{P({}^{1}_{10}\overline{x}', f({}^{1}_{8}\underline{y}')) \rightarrow\ {}^{1}_{1}\overline{y} \approx f({}^{1}_{1}\overline{y})\ {}^{1}_{5} \supset P({}^{1}_{4}\underline{x}, {}^{1}_{1}\overline{y})} \qquad \text{(by } (\neg \rightarrow))$$

$$\frac{}{P({}^{1}_{10}\overline{x}', f({}^{1}_{8}\underline{y}')),\ {}^{1}_{1}\overline{y} \approx f({}^{1}_{1}\overline{y}) \rightarrow P({}^{1}_{4}\underline{x}, {}^{1}_{1}\overline{y})} \qquad \text{(by } (\Box))$$

$$\text{(by } (\rightarrow\supset))$$

It is obvious that Tr and the substitution $\sigma = \{{}^{1}_{4}\underline{x} \mapsto {}^{1}_{10}\overline{x'},\ {}^{1}_{8}\underline{y'} \mapsto {}^{1}_{1}\overline{y}\}$ satisfy the conditions of both Theorem 3 and Theorem 4. We again obtain the deducibility of $\rightarrow H^*$ in $\mathbf{KJ}^{\approx}$ and $\mathbf{KK}^{\approx}$, but already on the basis of Theorems 3 and 4.

Draw your attention to that the just given inference is purely propositional although the initial sequent $\rightarrow H^*$ contains quantifiers and quantifier rules should be applied in **KC** for their elimination.

Turning back to the formulas H_1^* and H_2^* from Example 1 and applying Theorems 3 and 4 for obvious modifications of Tr and σ, we can "confirm" the results on the deducibility of $\rightarrow H_1^*$ and $\rightarrow H_2^*$ given in Example 1.

Taking into consideration all the above-given theorems, we can obtain the soundness and completeness theorem for any of our calculi if, and only if, such theorems take place for its Gentzen or Kanger analogue. For example, we conclude that the validity of a closed formula F in classical (intuitionistic) logic with equality is equivalent to the deductibility of the sequent $\rightarrow {}^{1}[_{\nu}F]$ in $\mathbf{QK}^{\approx}$ ($\mathbf{QJ}^{\approx}$).

9. CONCLUSION

The research presented in this paper demonstrates how the attempts to satisfy the EA principles in logical proof search gave rise to introducing the original notions of admissibility and compatibility, which is a good enough decision of the problem of the optimization of selecting such an order of quantifier rule applications that leads to the success in inference search in the fastest way in some sense of this word. As a result, a number of the quantifier-rule-free sequent calculi were constructed. They replace the step-by-step examination of all possible orders of quantifier rules applications by checking the conditions of admissibility and compatibility.

Proof search in such quantifier-rule-free calculi can be divided into four steps: propositional deduction, generating a substitution, checking admissibility and compatibility, applying equality rules if necessary. The notions of admissibility and compatibility permitted to replace quantifier rules applications by the certain technique for handling bound variables. Hence, further increasing of the efficiency of proof search in these calculi can be achieved by using, studying, and developing special (for example, goal-oriented) methods for propositional deduction in a concrete logic, technique for generating a necessary substitution (applying, for example, the results of the unification theory), tools for equality handling (developing, for example, a paramodulation and/or E-unification techniques).

Additionally note that the admissibility introduced in the paper has been incorporated into the classical logic "engines" of both the Russian and English SAD systems and demonstrated the good results.

References

[1] V.M. Glushkov. On some problems of computer technology and related mathematical problems. *Ukrainian Mathematical Journal*, 9(4): 369-376, 1957. In Russian.

[2] V.M. Glushkov. Some problems in the theories of automata and artificial intelligence. *Cybernetics and System Analysis*, 6(2), Springer, New York, 17-27, 1970.

[3] R. Boyer et al. The QED manifesto. *Lecture Notes in Artificial Intelligence*, 814: 238-251, 1994.

[4] The Mizar project. Web site at *http://www.mizar.org/project/*.

[5] The Calculemus project: Systems for integrated computations and deduction. Web site at *http://www.calculemus.net/*.

[6] A.V. Lyaletski and M.K. Morokhovets. Evidential paradigm: A current state. *Abstracts of AMS National Meeting "Mathematical Challenges of the 21st century"*. USA, Los Angeles, 81, 7-12 August, 2000.

[7] Yu. V. Kapitonova, K. P. Vershinin, A. I. Degtyarev, A. P. Zhezherun, and A. V. Lyaletski. System for processing mathematical texts. *Cybernetics and System Analysis*, 15(2): 209–210, Springer, New York, 1979.

[8] A. Lyaletski, K. Verchinine, A. Degtyarev, and A. Paskevich. System for Automated Deduction (SAD): linguistic and deductive peculiarities. *Intelligent Information Systems, 11th International Symposium, IIS 2002, Sopot, Poland, June 2002*, Advances in Soft Computing, 413-422, 2003.

[9] A. Lyaletski, M. Morokhovets, and A. Paskevich. Kyiv school of automated theorem proving: a historical chronicle. *Logic in Central and Eastern Europe: History, Science, and Discourse.* University Press of America, 431-469, 2012.

[10] V. M. Glushkov, Yu. V. Kapitonova, A. A Letichevskii, K. P. Vershinin, and N. P. Malevanyi. Construction of a practical formal language for mathematical theories. *Cybernetics and Systems Analysis*, 8(5): 730-739, 1972.

[11] , K. Vershinin and A. Paskevich, ForTheL — the language of formal theories, *International Journal of Information Theories and Applications*, 7(3): 120-126, 2000.

[12] Alexander Lyaletski and Mykola Nikitchenko. Logical and Semantic Investigations in Kyiv School of Automated Reasoning. *Trends in Contemporary Computer Science*, Bialystok University of Technology Publishing Office, Poland, 83-98, 2014.

[13] F. V. Anufriev, V. V. Fedyurko, A. A. Letichevskii, Z. M. Asel'derov, and I. I. Didukh. An algorithm for proving theorems in group theory. *Cybernetics and System Analysis*, 2(1), Springer, New York,20-25, 1966.

[14] F. V. Anufriev. An algorithm for proving theorems in set theory. *Theory of Automata*, volume 4, GIC AS UkrSSR, Kiev, 3-24, 1967. In Russian.

[15] F. V. Anufriyev. An algorithm of search for proofs of theorems in logical calculi. *Theory of Automata*, GIC AS UkrSSR, Kiev, 3-26, 1969. In Russian.

[16] A. I. Malashonok. The soundness and completeness of the Evidence Algorithm. *Computer-Aided Theorem Proving in Mathematics*, GIC AS UkrSSR, Kiev, 75-95, 1974. In Russian.

[17] S. Kanger. Simplified proof method for elementary logic. *Comp. Program. and Form. Sys.*, 87–93, 1963.

[18] A. I. Degtyarev and A. V. Lyaletski. Logical inference in the system for automated deduction, SAD. *Mathematical Foundations of Artificial Intelligence Systems*, GIC AS UkrSSR, Kiev, 3-11, 1981. In Russian.

[19] A. Degtyarev, A. Lyaletski, and M. Morokhovets. Evidence algorithm and sequent logical inference search. *LNAI (Proceedings of the International Conference LPAR'99)*, 1705: 44–61, 1999.

[20] A. Lyaletski and A. Paskevich. Goal-driven inference search in classical propositional logic. *Proc. International Workshop STRATEGIES'2001*, Siena, Italy, 65–74, 2001.

[21] T. Skolem. Logisch-kombinatorische Untersuchungen uber die Erfullbarkeit oder Beweisbarkeit mathematischer Satze. *Skriftner utgit ar Videnskapsselskaper i Kristiania*, 4: 4-36, 1920.

[22] J. Herbrand. Recherches sur la theorie de la demonstration. *Travaux de la Societe des Sciences et de Lettres de Varsovie, Class III, Sciences Mathematiques et Physiques*,33, 1930.

[23] G. Gentzen. Untersuchungen uber das logische schliessen (I, II). *Math. Zeit.*, 39: 176–210, 405–443, 1934.

[24] J. A. Robinson and A. Voronkov, editors. *Handbook of Automated Reasoning* (in 2 volumes), Elsevier and MIT Press, 2001.

[25] R. Hahnle. Tableaux and related methods. *Handbook of Automated Reasoning* (Edited by A.Robinson and A.Voronkov), Elsevier Science Publishers, 101-178, 2001.

[26] A. Degtyarev and A. Voronkov. Equality reasoning and sequent-based calculi. *Handbook of Automated Reasoning* (Edited by A.Robinson and A.Voronkov), Elsevier Science Publishers, 609-704, 2001.

[27] R. Nieuwenhuis and A. Rubio A. Paramodulation-based theorem proving. *Handbook of Automated Reasoning* (Edited by A.Robinson and A.Voronkov), Elsevier Science Publishers, 371-444, 2001.

[28] R. Letz and G. Stenz. Model elimination and connection tableau procedures. *Handbook of Automated Reasoning* (Edited by A.Robinson and A.Voronkov), Elsevier Science Publishers, 2017–2116, 2001.

[29] A. Voronkov. Proof search in intuitionistic logic based on constraint satisfaction and related complexity problems. *L.J. of the IGPL*, Oxford University Press, 1996.

[30] W. Bibel. *Automated Theorem Proving*. Vieweg Verlag, 1987.

[31] L. Wallen. *Automated proof search in non-classical logics.*, Oxford University Press, 1990.

[32] J. Otten and C. Kreitz. Connection-based theorem proving in classical and non-classical logics. *J. UCS*, 5(3): 88–112, 1999.

[33] J. Otten and C. Kreitz. A connection based proof method for intuitionistic logic. *Lecture Notes in Computer Science*, 918: 122–137, 1995.

[34] J. Otten and C. Kreitz. A uniform proof procedure for classical and non-classical logics. *Lecture Notes in Computer Science*,1137: 307–319, 1996.

[35] A. Waaler and L. Wallen. Tableaux for intuitionistic logics, *Handbook of Tableau Methods*, 255-296, 1999.

[36] A. Waaler and R. Antonsen. A free variable sequent calculus with uniform variable splitting, *Lecture Notes in Computer Science*, 2796: 214–229, Springer, 2003.

[37] A. Lyaletski. Variant of Herbrand theorem for formulas in prefix form. *Cybernetics*, 1: 12–116, 1981. In Russian.

[38] B. Konev and A. Lyaletski. Tableau proof search with admissible substitutions. *Proceedings of the International Workshop on First-order Theorem Proving*, Koblenz, Germany, 35-50, 2005.

[39] O. Hiroakira. Proof-theoretic methods in non-classical logic — an introduction. *Theories of Types and Proofs (M. Takahashi and M. Okada and M. Dezani-Ciancaglini, Eds.)*, Mathematical Society of Japan, Tokyo, 207-254, 1998.

[40] J. Robinson. A machine-oriented logic based on resolution principle, *J. of the ACM*, 12(1): 23-41, 1965.

[41] G. Mints. Herbrand theorem. *Mathematical Theory of Logical Inference*, 311-350, Nauka, Moscow, 1967. In Russian.

[42] J.H. Gallier. *Logic for Computer Science: Foundations of Automatic Theorem Proving*, Harper & Row, 1986.

[43] S. Autexier, H. Mantel, and W. Stephan. Simultaneous quantifier elimination, *Lecture Notes in Computer Science*, 1504: 141-152, 1998.

[44] M. Baaz and R. Iemhoff. The Skolemization of existential quantifiers in intuitionistic logic, *Annals of Pure and Applied Logic*, 142 (1): 269–295, 2006.

[45] B.B. Lewis. *Decidable Fragments and Skolemization in Superintuitionistic and Modal Logics*, Undergraduate Honors Thesis, Stanford University, 2012.

[46] A. Lyaletski.: Sequent forms of Herbrand theorem and their applications. *Annals of Mathematics and Artificial Intelligence*, 46(1-2): 191–230, Springer, 2006.

[47] A. Lyaletski. On some problems of efficient inference search in first-order cut-free modal sequent calculi. *Proceeding of the International Symposium SYNASC 2008*, 39-46, 2008.