Goal-Driven Inference Search in Classical Propositional Logic

Alexander Lyaletski and Andrey Paskevich

Cyberetics Department, Taras Shevchenko Kyiv National University,
2, Glushkov avenue, building 6, 03022 Kyiv, Ukraine,
tel.: +38 (044) 266 3108, fax: +38 (044) 266 1059,
e-mail: lav@tc.unicyb.kiev.ua, andrey@raptor.kiev.ua

Abstract. Two goal-driven strategies for inference search in propositional logic are presented in the form of special sequent calculi. Some results on their soundness and completeness are given, and comparison of these calculi is made. It is shown that no one of them is preferable to another in the sense of constructing minimal inferences.

1 Introduction

A programme of automated theorem proving in mathematics called Evidence Algorithm (EA) was advanced by Academician V.Glushkov in his paper [1]. The programme EA paid much attention to (interactive) control of the strategies for inference search. The first heuristic routine proposed for EA [2] provided already this possibility. Afterwards, that routine was extended to a complete Kanger-type [3] calculus [4], thus becoming a machine-oriented version of the usual sequent calculus [5]. Because of absence of the skolemization, this calculus has significantly lower efficiency than resolution-type calculi [7]. In order to overcome this obstacle (without introducing the skolemization), an original notion of an admissible substitution [8] have been developed and then have been used in a so-called a-sequent calculus (AS calculus) [9]. This calculus became a core of a prover implemented on the ES-series computers as a part of a software system called SAD (System of Automated Deduction) [10].

In the paper [9] the description of the AS calculus had partially an algorithmic character. This deviation from the traditional Gentzen notation was corrected, when a number of the modifications of AS was constructed in the frame of the INTAS-project 96-0760 (see, for example, [11, 12]). These modifications possesses the following features: goal-oriented reduction of an assertion to be proven to a number of auxiliary assertions; special quantifier handling technique (without the skolemization), which permits to achieve a search efficiency comparable with the efficiency of methods requiring the preliminary skolemization; separation of a deduction from equality processing (equation solving); use of human-like proof methods such as applications of definitions and auxiliary propositions; flexible interactive search mode.

All the above features are discussed in [11, 12]. However, these papers do not contain reasons on positive or negative peculiarities of different strategies for the

AS calculus and its modifications. The present paper is intended to fill partially this gap and it concerns two "goal-driven" strategies. Note that the strategies proved to be essentially different even in the case of propositional logic. In this connection these strategies are convenient to define as two different a-sequent calculi. All considerations are restricted to the propositional level.

2 Main Definitions

We consider classical propositional logic with the following connectives: implication ($\supset$), disjunction ($\vee$), conjunction ($\wedge$), and negation ($\neg$). Below propositional variables, or atoms, are denoted by A, B, C, or D, propositional literals (i.e. atoms and their negations) are denoted by L or M, propositions (formulas) are denoted by F, G, P, or Q. All the letters can be subscripted.

The expression $F^\neg$ denotes the result of one-step carrying of the negation into a proposition F: $(P \supset Q)^\neg$ is $P \wedge \neg Q$, $(P \vee Q)^\neg$ is $\neg P \wedge \neg Q$, $(P \wedge Q)^\neg$ is $\neg P \vee \neg Q$, $(\neg P)^\neg$ is P, and $A^\neg$ is $\neg A$.

We define the *positive* ($P[F^+]$) *and negative* ($P[F^-]$) *occurrences* of a proposition F to a proposition P by the following:

I. $F[F^+]$ and $F[\bar{F}^-]$ always hold, where $\bar{F}$ denotes a proposition G if F is $\neg G$, and $\bar{F}$ denotes $\neg F$ otherwise. ($\bar{F}$ is called a *complementary* of F.)

II. If a proposition F occurs in a proposition P, then:

$P[F^+]$ implies $(\neg P)[F^-]$	$P[F^-]$ implies $(\neg P)[F^+]$
$P[F^+]$ implies $(P \wedge Q)[F^+]$	$P[F^-]$ implies $(P \wedge Q)[F^-]$
$P[F^+]$ implies $(Q \wedge P)[F^+]$	$P[F^-]$ implies $(Q \wedge P)[F^-]$
$P[F^+]$ implies $(P \vee Q)[F^+]$	$P[F^-]$ implies $(P \vee Q)[F^-]$
$P[F^+]$ implies $(Q \vee P)[F^+]$	$P[F^-]$ implies $(Q \vee P)[F^-]$
$P[F^+]$ implies $(P \supset Q)[F^-]$	$P[F^-]$ implies $(P \supset Q)[F^+]$
$P[F^+]$ implies $(Q \supset P)[F^+]$	$P[F^-]$ implies $(Q \supset P)[F^-]$

The main object of the calculi is an *a-sequent*, which is an expression of the form $[\Lambda]\ \Gamma \to \Pi$, where Λ, Γ, Π are sequences (possibly, empty) of propositions. The propositions from Γ are called *premises*, the propositions from Π are called *goals*, and the propositions from Λ are called *framed propositions*. Here we consider a-sequents containing *one goal* and *framed literals* only.

Below we formulate two a-sequent calculi and give some results on them. Both calculi are based on the following principle: when a goal to be proven "interacts" with some premise in an a-sequent, new goals are generated. In this connection we call these calculi goal-driven. In the first calculus (denoted here by GD-1), the "interacting" premise is removed from the a-sequent, and new premises are generated if needed. In the second calculus (denoted by GD-2), the "interacting" premise is preserved without generating any new premise.

When a proof of an initial a-sequent is searched in GD-1 (GD-2), an *inference tree* is constructed. At the beginning of a search process the tree consists of the initial a-sequent. The subsequent nodes of the inference tree are generated in accordance with the rules of GD-1 (GD-2). Inference trees grow "from top to bottom" in accordance with the order of inference rule applications. An inference tree is considered to be a *proof tree* if every leaf in it is an axiom. An initial a-sequent of a proof tree in GD-1 (GD-2) is called *deducible* in GD-1 (GD-2).

The following a-sequent is an axiom of the calculi GD-1 and GD-2:

$$[\ ] \rightarrow \#$$

Note that the main results concern the "reducibility" of both GD-1 and GD-2 to the sequent calculus **G** [6] (restricted to classical propositional logic and denoted here by **Gal**), and vice versa. The soundness and completeness of GD-1 and GD-2 are the consequences of this "reducibility", since the calculus **Gal** is sound and complete.

3 Calculus GD-1

Goal-Splitting Rules (GS):

$$(\rightarrow\supset)_1: \quad \frac{[\Lambda]\ \ \Gamma \rightarrow F \supset G}{[\Lambda]\ \ \Gamma, F \rightarrow G} \qquad (\rightarrow\supset)_2: \quad \frac{[\Lambda]\ \ \Gamma \rightarrow F \supset G}{[\Lambda]\ \ \Gamma, \neg G \rightarrow \neg F}$$

$$(\rightarrow\vee)_1: \quad \frac{[\Lambda]\ \ \Gamma \rightarrow F \vee G}{[\Lambda]\ \ \Gamma, \neg F \rightarrow G} \qquad (\rightarrow\vee)_2: \quad \frac{[\Lambda]\ \ \Gamma \rightarrow F \vee G}{[\Lambda]\ \ \Gamma, \neg G \rightarrow F}$$

$$(\rightarrow\wedge): \quad \frac{[\Lambda]\ \ \Gamma \rightarrow F \wedge G}{[\Lambda]\ \ \Gamma \rightarrow F \qquad [\Lambda]\ \ \Gamma \rightarrow G} \qquad (\rightarrow\neg): \quad \frac{[\Lambda]\ \ \Gamma \rightarrow \neg F}{[\Lambda]\ \ \Gamma \rightarrow F^{\neg}}$$

Auxiliary-Goal Rules (AG):

$$(\supset\rightarrow)_1: \quad \frac{[\Lambda]\ \ \Gamma, F[L^-] \supset G, \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, (\neg F)[L^+], \Delta \rightarrow L \qquad [\Lambda, L^{\neg}]\ \ \Gamma, \Delta \rightarrow \neg G}$$

$$(\supset\rightarrow)_2: \quad \frac{[\Lambda]\ \ \Gamma, F \supset G[L^+], \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, G[L^+], \Delta \rightarrow L \qquad [\Lambda, L^{\neg}]\ \ \Gamma, \Delta \rightarrow F}$$

$$(\vee\rightarrow)_1: \quad \frac{[\Lambda]\ \ \Gamma, F[L^+] \vee G, \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, F[L^+], \Delta \rightarrow L \qquad [\Lambda, L^{\neg}]\ \ \Gamma, \Delta \rightarrow \neg G}$$

$$(\vee\rightarrow)_2: \quad \frac{[\Lambda]\ \ \Gamma, F \vee G[L^+], \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, G[L^+], \Delta \rightarrow L \qquad [\Lambda, L^{\neg}]\ \ \Gamma, \Delta \rightarrow \neg F}$$

$$(\wedge\rightarrow)_1: \quad \frac{[\Lambda]\ \ \Gamma, F[L^+] \wedge G, \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, F[L^+], G, \Delta \rightarrow L}$$

$$(\wedge\rightarrow)_2: \quad \frac{[\Lambda]\ \ \Gamma, F \wedge G[L^+], \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, F, G[L^+], \Delta \rightarrow L}$$

$$(\neg\rightarrow): \quad \frac{[\Lambda]\ \ \Gamma, \neg(F[L^-]), \Delta \rightarrow L}{[\Lambda]\ \ \Gamma, F^{\neg}[L^+], \Delta \rightarrow L}$$

Termination-by-Premise Rule (TP):
$$\frac{[\Lambda]\ \ \Gamma, L, \Delta \to L}{[\ ]\ \to \#}$$

Termination-by-Framed-Literal Rule (TF):
$$\frac{[\Lambda_1, L, \Lambda_2]\ \ \Gamma \to L}{[\ ]\ \to \#}$$

Remark. Whenever an AG-rule is applied, the corresponding positive occurrence of a goal literal must be fixed, so that the next AG-rule application is considered to refer to a premise containing the fixed occurrence. This means that any GD-1-inference cannot contain, for example, the "chain":

$$\frac{\dfrac{[\Lambda]\ (L \vee M) \wedge F, L \wedge G \to L}{[\Lambda]\ L \vee M, F, L \wedge G \to L}}{[\Lambda]\ L \vee M, F, L, G \to L}$$

because the last AG-rule application is made w.r.t. the formula $L \wedge G$, which does not contain the occurrence of L fixed in the previous AG-rule application (this fixed occurrence of L contains the formula $L \vee M$). The proof of Prop. 2 makes use of this remark.

Proposition 1. *If an a-sequent* $[L_1, \ldots, L_k]\ P_1, \ldots, P_n \to G$ *is deducible in GD-1, then the sequent* $L_1, \ldots, L_k, P_1, \ldots, P_n \to G$ *is deducible in* ***Gal****.*

Proof (scheme). First we prove that all the inference rules of **Gal** are "reversible", i.e. applying any rule to a deducible sequent, we generate deducible sequent(s). Then we show that carrying the negation in some formula of a sequent also does not affect its deducibility in **Gal**. Afterwards, it's easy to prove that every rule of GD-1 corresponds to a certain "superposition" of steps that preserve the deducibility in **Gal**.

Proposition 2. *Let* $P_1, \ldots, P_n$ *be a consistent set of propositions and* G *be a proposition. If the sequent* $P_1, \ldots, P_n \to G$ *is deducible in* ***Gal****, then the a-sequent* $[\ ]\ P_1, \ldots, P_n \to G$ *is deducible in GD-1.*

Proof (scheme). We prove the theorem by induction in the number of binary connectives in an initial sequence. There are two main cases.

I. The formula G is not a literal. In this case we generate deducible a-sequent(s) after applying an appropriate GS-rule (since the rules of **Gal** can be applied in any order without losing the deducibility).

II. The formula G is a literal. In this case we prove that there always exists some positive occurrence of G in premises of an a-sequent under consideration such that a deducible a-sequent(s) is generated after applying an appropriate AG-rule.

Corollary 1 (soundness and completeness of GD-1). *A proposition* F *is valid in classical propositional logic if and only if the a-sequent* $[\ \ ]\ \to F$ *is deducible in GD-1.*

Proof. This assertion follows from Prop. 1 and 2 and from soundness and completeness of **Gal**.

4 Calculus GD-2

Goal-Splitting Rules (GS):

$(\rightarrow\supset)_1 : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow F \supset G}{[\Lambda]\ \Gamma \rightarrow G}$ $\qquad (\rightarrow\supset)_2 : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow F \supset G}{[\Lambda]\ \Gamma \rightarrow \neg F}$

$(\rightarrow \vee)_1 : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow F \vee G}{[\Lambda]\ \Gamma \rightarrow G}$ $\qquad (\rightarrow \vee)_2 : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow F \vee G}{[\Lambda]\ \Gamma \rightarrow F}$

$(\rightarrow \wedge) : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow F \wedge G}{[\Lambda]\ \Gamma \rightarrow F \qquad [\Lambda]\ \Gamma \rightarrow G}$ $\qquad (\rightarrow \neg) : \quad \dfrac{[\Lambda]\ \Gamma \rightarrow \neg F}{[\Lambda]\ \Gamma \rightarrow F^{\neg}}$

Auxiliary-Goal Rule (AG): $\quad \dfrac{[\Lambda]\ \Gamma, F[L^{+}], \Delta \rightarrow L}{[\Lambda, L^{\neg}]\ \Gamma, F, \Delta \rightarrow \neg F}$

Termination-by-Premise Rule (TP): $\quad \dfrac{[\Lambda]\ \Gamma, L, \Delta \rightarrow L}{[\ ] \rightarrow \#}$

Termination-by-Framed-Literal Rule (TF): $\quad \dfrac{[\Lambda_1, L, \Lambda_2]\ \Gamma \rightarrow L}{[\ ] \rightarrow \#}$

Remark. Whenever an AG-rule is applied, the corresponding positive occurrence of a goal literal must be fixed, so that the next GS-rule applications must never remove this fixed occurrence from a proof. This means that any GD-2-inference cannot contain, for example, the "chain":

$$\cfrac{\cfrac{[\Lambda]\ L \wedge M \rightarrow L}{\overline{[\Lambda, \neg L]\ L \wedge M \rightarrow \neg L \vee \neg M}}}{[\Lambda, \neg L]\ L \wedge M \rightarrow \neg M}$$

because the last GS-rule removes $\neg L$ and preserves $\neg M$, though it must preserve $\neg L$. The proof of Prop. 4 makes use of this remark.

Proposition 3. *If an a-sequent $[L_1, \ldots, L_k]\ P_1, \ldots, P_n \rightarrow G$ is deducible in GD-2, then the sequent $L_1, \ldots, L_k, P_1, \ldots, P_n \rightarrow G$ is deducible in **Gal**.*

Proof (scheme). Using the proof of the Prop.1, we can show that adding instances of the same formula in a sequent does not affect to the deducibility of the sequent in the calculus **Gal**. Then we show that every rule of GD-2 corresponds to a certain "superposition" of steps that preserve the deducibility in **Gal**.

Proposition 4. *Let $P_1, \ldots, P_n$ be a consistent set of propositions and G be a proposition. If the sequent $P_1, \ldots, P_n \rightarrow G$ is deducible in **Gal** then the a-sequent $[\ ]\ P_1, \ldots, P_n, \neg G \rightarrow G$ is deducible in GD-2.*

Proof (scheme). We show that every inference in GD-1 can be transformed to an inference in GD-2 by the following way. Whenever we apply an AG-rule of GD-1 to a premise P, which is not an initial premise (i.e. it is derived), we choose one of the initial premises and split it up to P in GD-2. Since **Gal** is reducible to GD-1 (by Prop. 2), it is reducible to GD-2.

Corollary 2 (soundness and completeness of GD-2). *A proposition F is valid in classical propositional logic if and only if the a-sequent $[\;]\,\neg F \to F$ is deducible in GD-2.*

Proof. This assertion follows from Prop. 3 and 4 and from the soundness and completeness of **Gal**.

Remark. Note that some GS-rule applications in GD-2 remove subformulas from "parental" a-sequents. Therefore, we must insert $\neg F$ as the premise of the initial a-sequent in order to preserve the completeness (cp. Exam. 2).

5 Examples of Inferences

Here we give the examples of the proof trees $\mathbf{T}$ and $\mathbf{T}'$ in both calculi. We write an applied inference rule to the right of each step.

Example 1. Let us prove the a-sequent $[\;]\;A \vee B, A \supset B \to (B \supset C) \supset C$ in GD-1:

$\mathbf{T}$ is

$$\dfrac{\dfrac{[\;]\,A \vee B, A \supset B \to (B \supset C) \supset C}{[\;]\,A \vee B, A \supset B, B \supset C \to C}\,(\to\supset)_1}{\dfrac{[\;]\,A \vee B, A \supset B, C \to C}{[\;] \to \#}\,(\mathrm{TP}) \qquad \dfrac{[\neg C]\,A \vee B, A \supset B \to B}{\dfrac{[\neg C]\,A \vee B, B \to B}{[\;] \to \#}\,(\mathrm{TP}) \qquad \mathbf{T}_1}\,(\supset\to)_2}\,(\supset\to)_2$$

$\mathbf{T}_1$ is

$$\dfrac{[\neg C, \neg B]\,A \vee B \to A}{\dfrac{[\neg C, \neg B]\,A \to A}{[\;] \to \#}\,(\mathrm{TF}) \qquad \dfrac{[\neg C, \neg B, \neg A] \to \neg B}{[\;] \to \#}\,(\mathrm{TF})}\,(\vee\to)_1$$

Example 2. Let us prove the a-sequent $[\;]\;\Gamma \to A \vee D$ in GD-2, where the premises of Γ are $A \vee C, C \supset D, \neg(A \vee D)$:

$\mathbf{T}'$ is

$$\dfrac{\dfrac{\dfrac{\dfrac{[\;]\,\Gamma \to A \vee D}{[\;]\,\Gamma \to D}\,(\to\vee)_2}{[\neg D]\,\Gamma \to \neg(C \supset D)}\,(\mathrm{AG})}{[\neg D]\,\Gamma \to C \wedge \neg D}\,(\to\neg)}{\dfrac{\dfrac{[\neg D]\,\Gamma \to C}{[\neg D, \neg C]\,\Gamma \to \neg(A \vee C)}\,(\mathrm{AG})}{\mathbf{T}'_1}\,(\to\neg) \qquad \dfrac{[\neg D]\,\Gamma \to \neg D}{[\;] \to \#}\,(\mathrm{TF})}\,(\to\wedge)$$

$$\mathbf{T'}_1 \text{ is } \qquad \dfrac{[\neg D, \neg C]\; \Gamma \to \neg A \wedge \neg C}{\dfrac{\dfrac{\dfrac{\dfrac{[\neg D, \neg C]\; \Gamma \to \neg A}{[\neg D, \neg C, A]\; \Gamma \to \neg\neg(A \vee D)}\,(\mathrm{AG})}{[\neg D, \neg C, A]\; \Gamma \to A \vee D}\,(\to \neg)}{[\neg D, \neg C, A]\; \Gamma \to A}\,(\to \vee)_1}{[\;]\; \to \#}\,(\mathrm{TF}) \qquad \dfrac{[\neg D, \neg C]\; \Gamma \to \neg C}{[\;]\; \to \#}\,(\mathrm{TF})}\,(\to \wedge)$$

Note that we use both subformulas of the initial goal $A \vee D$. Thus, it was necessary to have its negation in Γ.

6 Comparison of the Calculi

When an inference search is made in GD-1 or GD-2, we apply subsequently the GS-rules until the goal of an a-sequent becomes a literal proposition.

When the goal becomes a literal (let us denote it by L), first of all we try to find it among the framed literals of the a-sequent in order to apply the termination rule (TF). If the TF-rule cannot be applied, we choose some premise P containing a positive occurrence of L. If P is L, we apply the termination rule (TP) and derive the axiom. In the opposite case we act in accordance with the calculus under consideration (GD-1 or GD-2).

In the case of GD-1 we subsequently apply the AG-rules until P is split up to L. When we do this, new branches can appear in an inference tree (with a new proposition in the goal).

In the case of GD-2 we apply the AG-rule, which does not split premise P, but writes its negation as a goal (preserving P as a premise). The complementary of L becomes a framed literal.

The main difference between the calculi is that GD-2 allows us to use initial premises many times in the same branch of an inference tree, while GD-1 keeps subformulas, which were not used in the deduction and can be used later.

The following examples demonstrate this difference. We define the *weight of a proof tree* as the number of a-sequents in it. A proof tree of some a-sequent is considered to be *minimal* if and only if it has the minimal weight among other proof trees for this a-sequent.

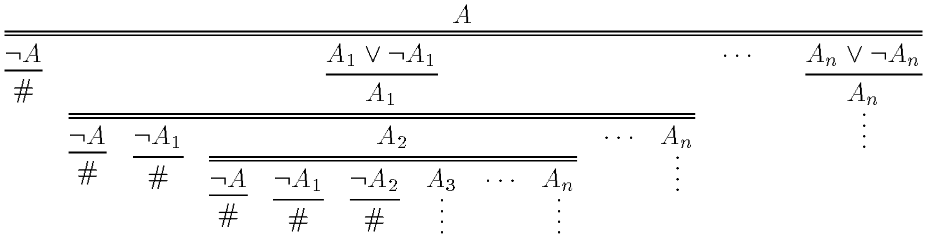
Example 3. Let us consider a-sequents of the form:

$$[\;]\; (A_1 \wedge \neg A_1) \vee \cdots \vee (A_n \wedge \neg A_n) \vee A \to A$$

Below we give the scheme of their inference in GD-1 (some steps are omitted).

$$\dfrac{[\;]\; (A_1 \wedge \neg A_1) \vee \cdots \vee (A_n \wedge \neg A_n) \vee A \to A}{\dfrac{[\;]\; A \to A}{[\;]\; \to \#} \qquad \dfrac{\dfrac{[\neg A]\; \to A_1 \vee \neg A_1}{[\neg A]\; A_1 \to A_1}}{[\;]\; \to \#} \qquad \cdots \qquad \dfrac{\dfrac{[\neg A]\; \to A_n \vee \neg A_n}{[\neg A]\; A_n \to A_n}}{[\;]\; \to \#}}$$

Let us construct the proof tree scheme for these a-sequents in GD-2. Since we do not change the set of premises during inference search, and since framed literals of any a-sequent in an inference tree are just complementaries of literal-goals of parent a-sequents, it will be sufficient to give the goals of a-sequents only:

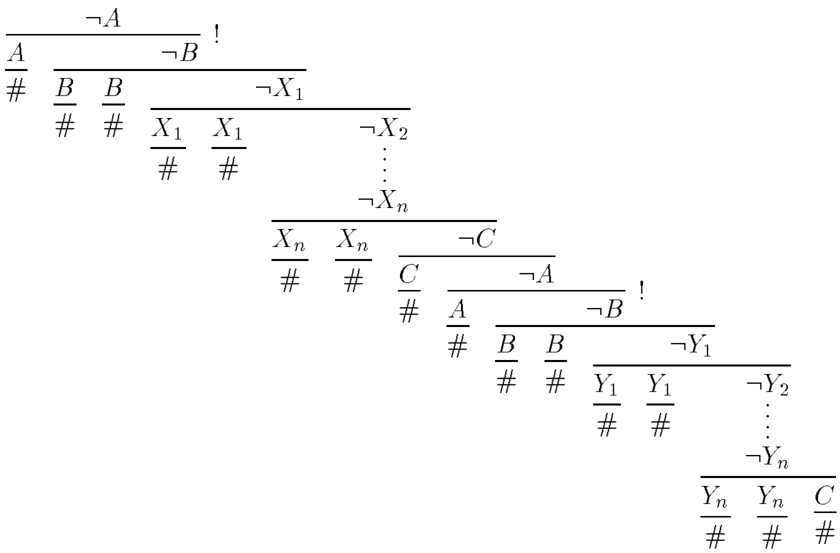


We see that weights of the corresponding minimal proof trees are linear by n in GD-1 and are exponential by n in GD-2 for the given class of a-sequents.

Example 4. Let us consider a-sequents of the form:

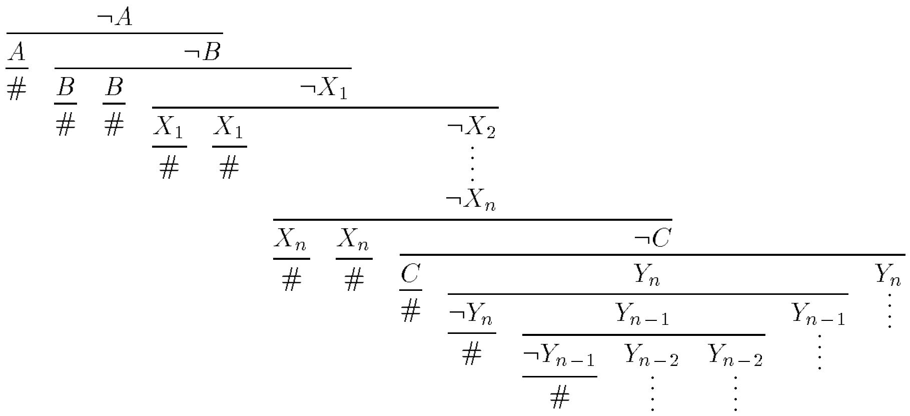
$$[\ ]\ B \supset (B \supset X_1),\ X_1 \supset (X_1 \supset X_2),\ \ldots,\ X_n \supset (X_n \supset C),\ C \supset A,$$
$$B \supset (B \supset Y_1),\ Y_1 \supset (Y_1 \supset Y_2),\ \ldots,\ Y_n \supset (Y_n \supset \neg C),\ A \supset B\ \rightarrow\ \neg A$$

Below we give the scheme of their inference in GD-2.



Note that the premise $A \supset B$ is used twice in the branch (the corresponding steps are marked by '!').

When deducing this a-sequent in GD-1, we pass the first half of the proof in the same manner. However, we cannot use again the premise $A \supset B$. Therefore, we must "inverse" the refutation of the literal C:



This a-sequent can be deduced in GD-1 in several ways, but no one of them gives a smaller inference. Thus, corresponding minimal weights are linear by n in GD-2 and are exponential by n in GD-1 for the given class of a-sequents.

7 Conclusion

The end of the 1950s and the beginning of the 1960s can be characterized as the period of coming into existence computers with a sufficiently high performance rate and information capacity. In response to this computing progress, a series of papers appeared, which contained the issues of implementation for Gentzen [5] calculi and for inference search methods relying on results of Skolem [13] and Herbrand [14]. For more detail, see, for example, [15].

Those first papers gave an answer to the principal question about a possibility to use computers for mathematical reasoning, but they did not contain special theorem-proving methods for constructing proofs of a rather high efficiency.

Investigations on improving the efficiency of theorem proving technique resulted in coming into existence Kanger's calculus [3] (Gentzen approach) and Robinson's resolution method [7] (Skolem-Herbrand approach). For those days, the latter approach yielded the most efficient machine-oriented inference search technique for 1st-order classical logic. As a result, studies on automated theorem proving were later concentrated upon improving the potentialities of the Skolem-Herbrand approach mainly. Thus, subsequent attempts for the advancement of the Skolem-Herbrand approach resulted in arising such methods as tableaux methods, the connection graph method, goal-oriented search methods, etc.

The lower efficiency of the Gentzen approach (as compared with the Skolem-Herbrand approach) can be explained by the fact that even Kanger-type calculi result in arising the superfluous enumeration caused by unordered logical rule applications and by the absence of a preliminary skolemization. At the same time, machine-oriented Kanger-type methods reflect proof technique that is more "natural" for a human. Also, it enables to control automated reasoning strategies and to construct rather flexible interactive search modes. Exactly these reasons caused the appearance of different modifications of AS.

All the modifications of AS use goal-driven approach, but only this paper contains some analysis how the choice of propositional rules has an influence on the efficiency of an inference search in sequent calculi. The above comparison of the calculi GD-1 and GD-2 shows that no one of them can be preferred to another, though GD-1 could seem to give always smaller minimal inferences than GD-2. This witnesses to the fact that even syntax of the problem under consideration influences essentially the choice of an appropriate strategy.

The authors thank referees for useful advice.

References

1. Glushkov,V.M.: Some problems of automata theory and artificial intelligence (in Russian). Kibernetika **2** (1970) 3–13
2. Anufriyev, F.V., Fediurko, V.V., Letichevski, A.A., Asel'derov, Z.M, Didukh, I.I.: On one algorithm of theorem proof search in Group Theory (in Russian). Kibernetika **1** (1966) 23–29
3. Kanger, S.: Simplified proof method for elementary logic. In: Comp. Program. and Form. Sys.: Studies in Logic, North-Holl. Publ. Co. (1963)
4. Anufriyev, F.V.: An algorithm of theorem proof search in logical calculi (in Russian). In: Teoriya avtomatov **5**, Institute of Cybernetics, Kiev (1969) 3–26
5. Gentzen, G.: Untersuchungen uber das Logische Schliessen. Math. Zeit. **39** (1934) 176–210
6. Gallier J.: Logic for Computer Science: Foundations of Automatic Theorem Proving. Harper and Row, Inc., New York (1986) 513 p.
7. Robinson, J.: A machine-oriented logic based on resolution principle. J. of the ACM (1965) 23–41
8. Lyaletski, A.V.: Gentzen calculi and admissible substitutions. In: Actes prelim. du Symp. Franco-Sovetigue "Informatika-91", Grenoble, France (1991)
9. Degtyarev, A.I., Lyaletski, A.V.: Logical inference in SAD (in Russian). In: Matematicheskiye osnovy sistem iskusstvennogo intellekta, Institute of Cybernetics, Kiev (1981)
10. Glushkov, V.M.: The System for Proving Automation (in Russian). In: Avtomatizatsiya obrabotki matematicheskikh tekstov, Institute of Cybernetics, Kiev (1980)
11. Degtyarev, A.I., Lyaletski, A.V., Morokhovets, M.K.: Evidence Algorithm and Sequent Logical Inference Search, In: Lect. Notes in Art. Intell. **1705** (1999) 44–61
12. Degtyarev, A.I., Lyaletski, A.V., Morokhovets, M.K.: On the EA-style integrated processing of self-contained mathematical texts. In: Proc. of the Intern. Workshop CALCULEMUS'2000, Great Britain (2000)
13. Skolem, T.: Logisch-kombinatorische Untersuchungen uber die Erfullbarkeit oder Beweisbarkeit mathematischer Satze. Skriftner utgit ar Videnskapsselskaper i Kristiania, **4** (1920) 4–36.
14. Herbrand, J.: Recherches sur la Theorie de la Demonstration. Travaux de la Societe des Sciences et de Lettres de Varsovie, Class III, Sciences Mathematiques et Physiques **33** (1930).
15. Chang, C., Lee, R.: Symbolic logic and mechanical theorem proving. Academic Press, New York (1973).