The Complexity of the Disjunction and Existential Properties in Intuitionistic Logic

Sam Buss*
Department of Mathematics
University of California, San Diego

Grigori Mints**
Department of Philosophy
Stanford University

February 5, 1999

Abstract

This paper considers the computational complexity of the disjunction and existential properties of intuitionistic logic. We prove that the disjunction property holds feasibly for intuitionistic propositional logic; i.e., from a proof of $A \vee B$, a proof either of A or of B can be found in polynomial time. For intuitionistic predicate logic, we prove superexponential lower bounds for the disjunction property, namely, there is a superexponential lower bound on the time required, given a proof of $A \vee B$, to produce one of A and B which is true. In addition, there is superexponential lower bound on the size of terms which fulfill the existential property of intuitionistic predicate logic. There are superexponential upper bounds for these problems, so the lower bounds are essentially optimal.

MSC CODES: 03F05, 03F20, 03F55, 03C40, 68Q15, 68N17.
KEYWORDS: intuitionistic logic, cut-elimination, Craig interpolation, polynomial-time, Horn resolution, proof complexity, natural deduction, induction speed-up.

1 Introduction

It is a well-known fact [1] that intuitionistic logic satisfies the following disjunction and existential properties: (throughout this paper, $\vdash$ represents intuitionistic provability.)

*Supported in part by NSF grant DMS-9503247 and grant INT-9600919/ME-103 from NSF and MŠMT (Czech Republic)
**On sabbatical leave from Stanford University. Supported in part by Rome Labs under contract F30602-97-C-0146.

- If $\vdash A \vee B$, then $\vdash A$ or $\vdash B$.
- If $\vdash (\exists x)A(x)$, then $\vdash A(t)$ for some term t.

We are interested in studying the problem of the complexity of the disjunction and existential properties. For instance, given that $A \vee B$ is intuitionistically valid, how hard is it to identify one of A or B as intuitionistically valid? Or, given an intuitionistically valid formula $(\exists x)A(x)$, how hard is it to find a term t such that $A(t)$ is intuitionistically valid? For the case of propositional logic, there is a PSPACE algorithm for the first problem, since propositional validity is decidable in polynomial space [3, 7]. For the second problem, if the first-order language contains only predicate symbols and constant symbols, but no function symbols, then the term t can be taken to be either a constant symbol or a free variable of $A(x)$ so there is a trivial constant upper bound on the size of the term t. However, if there is a non-unary function symbol in the language, then there is no recursive upper bound on the size of t (as a function of A). Likewise, in predicate logic with at least one non-unary function symbol, there is no recursive bound on the complexity of deciding the disjunction property (using the fact that there are r.e. sets which cannot be recursively separated).[1] From this last fact, one can see that in intuitionistic predicate logic, even with no function symbols, there is no recursive bound on the computational complexity of finding a term t that fulfills the existential property for a given valid formula $(\exists x)A(x)$; similarly the disjunction property can be non-recursive in theories without function symbols.[2]

The previous paragraph discussed complexity bounds in terms of the formulas $A \vee B$ or $(\exists x)A(x)$. It is, however, more interesting to determine

[1] Here is a quick sketch of the proof of these two assertions: let $U(e,t,x)$ be a formula expressing the condition that Turing machine with Gödel number e halts within t steps outputting $x \in \{0,1\}$. It is possible to formulate $U(e,t,x)$ as a first-order formula so that $\vdash U(\underline{e},\underline{t},\underline{x})$ holds whenever U is true of the values represented by $\underline{e},\underline{t},\underline{x}$; here $\underline{m}$ denotes the term $S^m 0$. Consideration of the formulas $\exists t \exists x U(\underline{e},t,x)$ which are true illustrates the fact that t cannot be recursively bounded in terms of the size of $\exists t \exists x U(\underline{e},t,x)$, since otherwise the halting problem would be decidable. To prove the non-recursiveness of the disjunction property, the set of formulas of the form

$$(\exists t)U(\underline{e},t,0) \vee (\exists t)U(\underline{e},t,1)$$

which are true shows that deciding on a valid disjunct cannot be a recursive process.

[2] If there are no function symbols, we can use relation symbols instead of function symbols and formulate $U_e^*(x)$ which asserts that Turing machine e eventually halts with output $x \in \{0,1\}$. Then the sentences $(\exists x)U_e^*(x)$ illustrate the non-recursiveness of the existential property, and the sentences $U_e^*(0) \vee U_e^*(1)$ illustrate the non-recursiveness of the disjunction property.

the computational complexity of solving the disjunction and existential properties assuming we are given a *proof* of $A \vee B$ or of $(\exists x)A$, respectively; and these are the questions we will address in this paper.

There are a variety of related questions here: for example, one could ask about (a) the complexity of determining a particular one of A or B to be valid, (b) bounds on the size of the shortest proof of either A or B, or (c) the computational complexity of producing a proof of either A or B; in each of the three cases assuming that a proof of $A \vee B$ is given as input. In propositional logic, we show below that there is a polynomial time algorithm which produces a proof of either A or B from a proof of $A \vee B$. This also gives a polynomial time bound for the problem (a) and a polynomial upper bound for the problem (b) for propositional intuitionistic logic. For predicate logic, we shall give an superexponential lower bound on the time required to recognize one of A or B as valid, even given a proof of $A \vee B$. This immediately implies also a superexponential lower bound on the size of a proof of either A or B and thereby a superexponential lower bound for the time required to produce such a proof.

For the existential property, we give a superexponential lower bound on the size of a term t such that $A(t)$ is intuitionistically valid; this superexponential bound is in terms of the size of a proof of $(\exists x)A(x)$.

The above superexponential lower bounds are easily seen to be essentially optimal, since a proof-normalization procedure can be used to solve the disjunction and existential properties.

For the special case of predicate logic with no function symbols, we give an exponential upper bound on the computational complexity of the disjunction and existential properties.

2 The propositional disjunction property

We formalize intuitionistic propositional calculus as a natural deduction system with explicit listing of assumptions. In this system, we use a *sequent* $A_1, \ldots, A_n \Rightarrow B \quad (n \geq 0)$ to denote the fact the formula B has been derived from the assumptions $A_1, \ldots, A_n$. The assumptions $A_1, \ldots, A_n$ form a set.

When we speak of a formula F being provable, we mean the sequent $\Rightarrow F$.

The axioms of our intuitionistic propositional logic are $A \Rightarrow A$ and $\bot \Rightarrow A$, for A any formula.

The inference rules are standard introduction and elimination rules for $\wedge$, $\vee$, and $\supset$. These are the inferences

$$\frac{\Gamma \Rightarrow A \wedge B}{\Gamma \Rightarrow A} \qquad \frac{\Gamma \Rightarrow A \wedge B}{\Gamma \Rightarrow B} \qquad \frac{\Gamma \Rightarrow A \quad \Delta \Rightarrow B}{\Gamma, \Delta \Rightarrow A \wedge B}$$

$$\frac{\Gamma \Rightarrow A \vee B \quad \Delta, A \Rightarrow C \quad \Pi, B \Rightarrow C}{\Gamma, \Delta, \Pi \Rightarrow C} \qquad \frac{\Gamma \Rightarrow A}{\Gamma \Rightarrow A \vee B} \qquad \frac{\Gamma \Rightarrow B}{\Gamma \Rightarrow A \vee B}$$

$$\frac{\Gamma \Rightarrow A \supset B \quad \Delta \Rightarrow A}{\Gamma, \Delta \Rightarrow B} \qquad \frac{\Gamma, A \Rightarrow B}{\Gamma \Rightarrow A \supset B}$$

The four rules in the lefthand side are called *elimination rules* and the rest are called *introduction rules*. A *cut* in a natural deduction proof consists of an introduction rule whose conclusion is the principal (i.e., leftmost) hypothesis of an elimination rule. It is well known that natural deduction proofs can be normalized, so that the cuts can be removed from natural deduction proofs [5]. When cuts are permitted, the natural deduction system is equivalent the to sequent calculus and to Hilbert-style systems in that proofs in one system can be converted into a proof in another system by a polynomial time algorithm.

For the rest of this section, we assume that an intuitionistic natural deduction proof d_0 of some formula is fixed.

Definition *A sequent S of the form $\Gamma \Rightarrow A$ is* immediately derivable (i.d.) *according to the following inductive definition:*

(a) *S occurs in d_0, or*

(b) *$C, \Gamma \Rightarrow A$ and $\Rightarrow C$ are both i.d., for some formula C.*

In other words, the i.d. sequents are obtained from sequents present in d_0 by sequent calculus style cuts with i.d. formulas.

Lemma 1 *Every immediately derivable sequent is derivable. Furthermore, there is a polynomial time algorithm which, given d_0, gives derivations of all its i.d. sequents.*

Proof It is obvious by the induction on the definition of i.d. sequents that every i.d. sequent is derivable. Furthermore, since the i.d. sequents are obtained only by sequent calculus style cuts from sequents in d_0, this is essentially the same as reasoning with Horn clauses using only SLD resolution, which is easily seen (and well-known) to be polynomially time complete. $\dashv$

The following statement provides a simple polynomial time method of finding a provable disjunct from a given proof d_0. Note that no normal form property is assumed for d_0.

Theorem 2 *If $d_0 :\Rightarrow A \vee B$ then at least one of A and B is i.d.*

As an immediate consequence of Lemma 1 and Theorem 2, we have:

Corollary 3 *There is a polynomial time algorithm which, given an propositional intuitionistic proof of $A \vee B$, produces a proof of either A or B.*

In order to prove Theorem 2 we shall use a restricted normalization process in which only certain cuts are eliminated. Recall that a cut (maximal formula) in a natural deduction is a conclusion of an introduction rule which is the principal formula of an elimination rule (i.e. it contains the connective to be eliminated). We call a cut *assumption-free* if its last sequent (the conclusion of the elimination rule) contains no assumption, i.e. is of the form $\Rightarrow F$. Let d_1 be the result of eliminating all assumption-free cuts from d_0 by the standard cut-reduction steps (this process is recalled below in the proof of Lemma 5). The proof d_1 exists since every sequence of cut-reductions terminates.

Lemma 4 *If a proof $d :\Rightarrow F$ does not contain assumption-free cuts, then it ends in an introduction rule.*

Proof Suppose that the proof does not end in an introduction rule. Then, it must end with one of the four elimination rules shown above. The last sequent has no assumptions, and therefore the leftmost hypothesis of the introduction rule also has no assumptions. Continue traversing upwards in the proof tree for as long as we encounter elimination rules, always choosing the leftmost branch. The sequents we reach in this traversal all have no assumptions and eventually we must arrive either at an introduction inference or at an initial sequent. However, it is impossible to arrive at an initial sequent, since all initial sequents have assumptions. Likewise, it is impossible to arrive at an introduction rule, since this would be an assumption-free cut and d has no assumption free cuts.

Therefore, we have obtained a contradiction, so d must end with an introduction rule. $\dashv$

Lemma 4 implies that the final inference of d_1 is an $\vee$-introduction, and thus either $\Rightarrow A$ or $\Rightarrow B$ is the penultimate sequent in d_1. To finish

the proof of the Theorem 2 it will suffice to show that every sequent in d_1 including the premise A or B of the concluding $\vee$-introduction is i.d. (with respect to the *original* proof d_0). This fact follows immediately from the next lemma.

Lemma 5 *If d converts to d' by a single reduction of an assumption-free cut, then every sequent i.d. with respect to d' is i.d. with respect to d.*

Proof It is certainly sufficient to verify only that every sequent appearing in d' is i.d. with respect to d, since the other clause of the definition of i.d. is obviously preserved. There are three cases to consider corresponding to possible reductions.

In the case of $\wedge$-reduction the proof is reduced according to:

$$\dfrac{\dfrac{\begin{matrix} d^* \\ \vdots \\ \Rightarrow A \end{matrix} \qquad \Rightarrow B}{\Rightarrow A \wedge B}}{\Rightarrow A} \qquad \text{reduces to} \qquad \begin{matrix} d^* \\ \vdots \\ \Rightarrow A \end{matrix}$$

Thus no new sequents appear in d' and the lemma holds trivially.

In the case of $\supset$-reduction the proof is reduced according to:

$$\dfrac{\dfrac{\begin{matrix} A \Rightarrow A \\ d^* \\ A \Rightarrow F \end{matrix}}{\Rightarrow A \supset F} \qquad \Rightarrow A}{\Rightarrow F} \qquad \text{reduces to} \qquad \begin{matrix} \Rightarrow A \\ d^{*\prime} \\ \Rightarrow F \end{matrix}$$

where there may be multiple occurrences of the axiom $A \Rightarrow A$ in the subproof d^*, and where $d^{*\prime}$ is the same as d' except with occurrences of A deleted from the assumptions of sequents. Since A was i.d. in d, each sequent in d' is i.d. with respect to d.

In the final case of $\vee$-reduction the proof is reduced according to:

$$\dfrac{\dfrac{\Rightarrow A}{\Rightarrow A \vee B} \qquad \begin{matrix} A \Rightarrow A \\ d^* \\ A \Rightarrow F \end{matrix} \qquad B \Rightarrow F}{\Rightarrow F} \qquad \text{reduces to} \qquad \begin{matrix} \Rightarrow A \\ d^{*\prime} \\ \Rightarrow F \end{matrix}$$

Again new sequents in d' are obtained from sequents in d by deleting occurrences of i.d. formula A from assumptions. So each sequent in d' is i.d. with respect to d. $\dashv$

That completes the proof of Theorem 2.

3 Upper bounds for predicate logic

In the following we deal with the intuitionistic predicate logic with arbitrary predicate, constant and function symbols formalized as a natural deduction calculus. The axioms and rules of inference for the natural deduction proof system for intuitionistic predicate logic are those of the above defined propositional system plus elimination and introduction rules for quantifiers:

$$\frac{\Gamma \Rightarrow (\forall x)A(x)}{\Gamma \Rightarrow A(t)} \qquad \frac{\Gamma \Rightarrow A(b)}{\Gamma \Rightarrow (\forall x)A(x)}$$

$$\frac{\Gamma \Rightarrow (\exists x)A(x) \qquad \Delta, A(b) \Rightarrow C}{\Gamma, \Delta \Rightarrow C} \qquad \frac{\Gamma \Rightarrow A(t)}{\Gamma \Rightarrow (\exists x)A(x)}$$

In these rules, the symbol b is a free variable, called the *eigenvariable* of the inference: it is required that b not occur free in the conclusion of the inference.

For predicate logic, we modify the definition of immediate derivability to allow substituting terms for (eigen)variables: Let a proof d_0 be fixed.

Definition *A sequent S is* immediately derivable (i.d.) *according to the following inductive definition:*

(a) *S occurs in d_0, or*

(b) *S is of the form $\Gamma \Rightarrow A$ and there is a formula C such that $C, \Gamma \Rightarrow A$ and $\Rightarrow C$ are both i.d.*

(c) *S is of the form $\Gamma(t) \Rightarrow A(t)$ where $\Gamma(b) \Rightarrow A(b)$ is i.d.*

The next lemma is immediate from the definition of i.d.

Lemma 6 *Every immediately derivable sequent is derivable*

Theorem 7

(a) *If d_0 is a proof of $\Rightarrow A \vee B$ then at least one of A and B is i.d.*

(b) *If d_0 is a proof of $(\exists x)A(x)$ then $A(t)$ is i.d. for some term t.*

Proof Theorem 7 is proved exactly like Theorem 2: we need only check the two additional reduction cases that now arise in the proof of Lemma 5.

In the case of a $\forall$-reduction the proof is reduced according to:

$$\dfrac{\dfrac{\begin{array}{c} d^* \\ \Rightarrow A(b) \end{array}}{\Rightarrow (\forall x)A(x)}}{\Rightarrow A(t)} \qquad \text{reduces to} \qquad \begin{array}{c} d^{*\prime} \\ \Rightarrow A(t) \end{array}$$

The subproof $d^{*\prime}$ is obtained from d^* by replacing all relevant occurrences of the variable b with the term t. Thus every sequent in the reduced proof d' is i.d. with respect to the original proof d.

In the case of an $\exists$-reduction the proof is reduced according to:

$$\dfrac{\dfrac{\Rightarrow A(t)}{\Rightarrow (\exists x)A(x)} \qquad \begin{array}{c} A(b) \Rightarrow A(b) \\ d^{*\prime} \\ A(b) \Rightarrow F \end{array}}{\Rightarrow F} \qquad \text{reduces to} \qquad \begin{array}{c} \Rightarrow A(t) \\ d^{*\prime} \\ \Rightarrow F \end{array}$$

Again, every sequent in the reduced proof d' is i.d. with respect to the original proof d. $\dashv$

Theorem 7 provides a simple description of the set of i.d. formulas. An obvious upper bound on the number of i.d. formulas is the total number of sequents in the proof d_1 obtained by elimination of assumption-free cuts from the original natural deduction d_0. It is well-known that there is (only) a superexponential blowup in the size of proofs during the normalization procedure. This gives a superexponential upper bound on the time complexity of eliminating assumption-free cuts from proof d_0. Therefore, there is a superexponential time algorithm which, given a proof d_0, produces a set $\mathcal{S}$ of sequents such that every i.d. sequent is a substitution instance of one of the sequents in $\mathcal{S}$. We call $\mathcal{S}$ a *complete set of i.d. sequents*.

More precisely, define $2 \Uparrow c$ by $2 \Uparrow 0 = 0$ and $2 \Uparrow (n+1) = 2^{2 \Uparrow n}$. Then we have established the following upper bound.

Theorem 8 *There is an algorithm with runtime $2 \Uparrow (cn)$, for some constant c, which upon input a proof of n symbols, produces proofs of a complete set of i.d. sequents for d_0.*

By Theorem 7, this superexponential time algorithm solves the disjunction and existential properties.

In the next section, we shall prove that this superexponential order of magnitude for the runtime is essentially optimal. However, in the special case of predicate logic with no function symbols, the terms t which are used in substituting into i.d. sequents according to the third case of the

definition of i.d., may be required (w.l.o.g.) to be one of the variables or constant symbols which appear in d_0. With this restriction on the terms t, there are only exponentially many possible i.d. formulas. Therefore, using the usual SLD resolution method for Horn clauses, there is an exponential time algorithm which produces a complete set of i.d. sequents for d_0. This establishes:

Theorem 9 *There is an exponential time algorithm which solves the disjunction and existential properties for intuitionistic predicate logic with no function symbols.*

4 The existential property lower bound

This section will describe the superexponential lower bound for the existential property for intuitionistic predicate logic. The lower bound is established by exhibiting a family of sentences $(\exists x)A(x)$ which have short, polynomial length proofs such that the formulas $A(t)$ are valid only for superexponentially long terms t.

By the previous two sections, any lower bound on the existential property must include also a lower bound on the number of steps in proof normalization; thus it is no surprise that our proof uses the 'induction speed-up' method of Solovay (which is similar to the much earlier construction which Gentzen used for the provability of transfinite induction) which happens also to be one of the best tools for proving lower bounds on proof normalization and on cut-elimination. The first uses of the induction speed-up method for proving such lower bounds were by Orevkov [4] and Statman [8]; see also the survey of Pudlak [6].

We will choose the predicate language containing the constant symbol 0, the unary function S, the infix binary function symbol $+$ and two binary predicate symbols $=$ and e. One can intuitively think of these symbols representing zero, successor, addition and equality and of $e(x, y)$ representing the relation $y = 2^x$. Define G to be the universal closure of the conjunction of the following formulas:

$$
\begin{array}{ll}
x + 0 = x & x = x \\
0 + x = x & x = y \supset y = x \\
x + Sy = S(x + y) & x = y \wedge y = z \supset x = z \\
(x + y) + z = x + (y + z) & x = y \supset Sx = Sy \\
e(0, S0) & x = y \wedge u = v \supset (x + u) = (y + v) \\
e(x, y) \supset e(Sx, y + y) & x = y \wedge u = v \wedge e(x, u) \supset e(y, v)
\end{array}
$$

The axioms in the righthand column are of course just the familiar equality axioms — we are assuming that our underlying predicate logic does not have the equality relation as a *logical* symbol, but only as a *non-logical* symbol.

We define the following formulas $J_m(x)$ and $K_m(x)$ by induction on m:

$$\begin{aligned} J_0(x) &\Leftrightarrow 0 = 0 \\ K_m(x) &\Leftrightarrow (\exists y)((G \supset e(x,y)) \wedge J_m(y)) \\ J_{m+1}(x) &\Leftrightarrow (\forall z)(K_m(z) \supset K_m(z+x)). \end{aligned}$$

It is easy to see that there are intuitionistic proofs of the following formulas:

$K_0(0)$

$(\forall x)(K_0(x) \supset K_0(Sx))$

$J_1(0)$

$(\forall x)(J_1(x) \supset J_1(Sx))$

$J_1(S0)$

$(\forall x)(J_1(x) \supset J_1(x+x))$

$(\forall x)(J_1(x) \supset K_0(x))$

Continuing by induction on m, the same constructions show there are proof of the following formulas, and that furthermore these proofs have size polynomial in m.

$K_m(0)$

$(\forall x)(K_m(x) \supset K_m(Sx))$

$J_{m+1}(0)$

$(\forall x)(J_{m+1}(x) \supset J_{m+1}(Sx))$

$J_{m+1}(S0)$

$(\forall x)(J_{m+1}(x) \supset J_{m+1}(x+x))$

$(\forall x)(J_{m+1}(x) \supset K_m(x))$

We thus immediately get polynomial size (in m) proofs of

$$(\forall x)(K_{m+1}(x) \supset (\exists y)((G \supset e(x,y)) \wedge K_m(x)))$$

and iterating this m times, polynomial size proofs of

$$(\forall x)(\exists y_m)\cdots(\exists y_1)[G \supset (e(x,y_1) \wedge e(y_1,y_2) \wedge \cdots \wedge e(y_{m-1},y_m))].$$

With $x = 0$, we get thereby also polynomial size proofs of the sentences

$$(\exists y_m)\cdots(\exists y_1)[G \supset (e(0,y_1) \wedge e(y_1,y_2) \wedge \cdots \wedge e(y_{m-1},y_m))]. \qquad (1)$$

Now consider the complexity of the existential property for this last intuitionistically derivable sentence. If we consider the standard (classical) model of the integers with zero, successor, addition, true equality and exponentiation, we see that the only terms that can be substituted into equation (1) for the variable y_m and yield a true formula are the terms with value $2 \Uparrow m$. Since the only function symbols at our disposal are successor and addition, any term with value $2 \Uparrow m$ must have at least $2 \Uparrow (m-1)$ symbols. On the other hand, equation (1) has an intuitionistic proof of $n = m^{O(1)}$ symbols.

Thus we have established that the existential property for intuitionistic predicate logic has superexponential (i.e., stack of twos of height n^{ϵ}) complexity.[3]

5 The disjunction property lower bound

In this section, the superexponential lower bound for the existential property is extended to a similar superexponential lower bound for the disjunction property. For the lower bound for the disjunction property, we cannot use just superexponential growth rate. Instead, we will use a Turing machine for which it is difficult to predict what state it will be in a future point in time. We shall pick a fixed Turing machine M which has a single, two-way infinite tape and has only the two alphabet symbols a and b. The machine M will have the property that the problem of, given m, determining which state M will be in after computing for exactly $2 \Uparrow m$ steps starting on a blank tape

[3] Our construction did not minimize the number of non-logical symbols in the language. With a more complicated version of the same proof, we could use a language with only a binary function symbol. As we remarked in section 3, the existential property for a first-order language with no function symbols has at worst exponential-time computational complexity.

(b is the blank symbol) is not in the complexity class $TIME(2 \Uparrow (m - c))$ for some constant c. Such a Turing machine can easily be shown to exist using the Hartmanis-Stearns time hierarchy theorem.

For convenience sake, we use a larger language for predicate logic, which, in addition to the symbols $=$, 0, S, $+$ and e, contains constants a and b for the tape symbols of M, constants $q_1, \ldots, q_s$ which represent the s states of M, a 4-ary predicate symbol ID, a binary function symbol σ, and a unary predicate symbol $Dfnt$. The intuitive idea of σ is that terms of the form $\sigma(c_1, \sigma(c_2, \sigma(c_3, \ldots)))$ represent strings $c_1c_2c_3\cdots$ where each c_i is either a or b. Then the predicate $ID(t, q, \alpha, \beta)$ is intended to mean that at time step t, M is in state q, with the string β to the right of the tape head, and with the string α to the left of the tape head. It is convenient to reverse the symbol order to the left of the tape head, so that in fact α represents the string obtained by starting with the symbol under the tape head and then moving leftward. "ID" stands for "instantaneous description." The intuitive idea for $Dfnt(x)$ is that x is the value of a term built up from a's, b's and σ's in some 'definite' or 'decidable' way.

The set G^* is defined to the conjunction of the universal closures of the following set of formulas:

(a) Every formula in G is included in G^*, in addition, the equality axioms for ID, σ and $Dfnt$ are included.

(b) $b = \sigma(b, b)$. Intuitively, this accounts for the fact that the tape is filled with blanks (b's) everywhere past the ends of the half-tapes.

(c) $ID(0, q_1, b, b)$. Intuitively, the machine M starts on a blank tape.

(d) For each transition rule (q_i, c, q_j, d) of M, where $c, d \in \{a, b\}$, G^* includes

$$ID(x, q_i, \sigma(c, u), v) \rightarrow ID(Sx, q_j, \sigma(d, u), v).$$

Intuitively: if in state q_i, reading c, then write d and go to state q_j.

(e) For each transition rule (q_i, c, q_j, L) of M, where $c \in \{a, b\}$ and "L" denotes "move left", G^* includes

$$ID(x, q_i, \sigma(c, u), v) \rightarrow ID(Sx, q_j, u, \sigma(c, v)).$$

Intuitively: if in state q_i, reading c, then move left one square and go to state q_j.

(f) For each transition rule (q_i, c, q_j, R) of M, where $c \in \{a, b\}$ and "R" denotes "move right", and for each $d \in \{a, b\}$, G^* includes

$$ID(x, q_i, \sigma(c, u), \sigma(d, v)) \rightarrow ID(Sx, q_j, \sigma(d, \sigma(c, u)), v).$$

Intuitively: if in state q_i, reading c, then move right one square and go to state q_j.

(g) G^* includes the formulas $Dfnt(b)$ and

$$Dfnt(x) \leftrightarrow (\exists u)(Dfnt(u) \wedge (x = \sigma(a, u) \vee x = \sigma(b, u))).$$

We let $qDfnt(x)$ abbreviate the formula

$$x = q_1 \vee x = q_2 \vee \cdots \vee x = q_s.$$

Next we define analogues $J_n^*(x)$ and $K_n^*(x)$ of the formulas $J_n(x)$ and $K_n(x)$ from the previous section. However, instead of starting the inductive definition with $J_0^*(x)$, the base definition is

$$K_{-1}^*(x) \Leftrightarrow (\exists q)(\exists u)(\exists v)[G \supset (ID(x, q, u, v) \wedge qDfnt(q) \wedge Dfnt(u) \wedge Dfnt(v))].$$

It is easy to give intuitionistic proofs of

$K_{-1}^*(0)$, and

$(\forall x)(K_{-1}^*(x) \supset K_{-1}^*(Sx))$

Then, by induction on $m \geq -1$, we define

$$\begin{aligned} J_{m+1}^* &\Leftrightarrow (\forall z)(K_m(z) \supset K_m(z + x)) \\ K_{m+1}^* &\Leftrightarrow (\exists y)((G \supset e(x, y)) \wedge J_{m+1}(y)) \end{aligned}$$

By arguments very similar to the ones given before, there are intuitionistic proofs of the following formulas

$K_m^*(0)$

$(\forall x)(K_m^*(x) \supset K_m^*(Sx))$

$J_{m+1}^*(0)$

$(\forall x)(J_{m+1}^*(x) \supset J_{m+1}^*(Sx))$

$J^*_{m+1}(S0)$

$(\forall x)(J^*_{m+1}(x) \supset J^*_{m+1}(x+x))$

$(\forall x)(J^*_{m+1}(x) \supset K^*_m(x))$

and the proofs of the above formulas are all polynomial size in m. We thus immediately get polynomial size proofs of

$$(\forall x)(K^*_{m+1}(x) \supset (\exists y)((G \supset e(x,y)) \wedge K^*_m(x)))$$

and iterating this m times, a polynomial size proof of

$$(\forall x)(\exists y_m)\cdots(\exists y_1)[K^*_{-1}(y_m) \wedge [G \supset (e(x,y_1) \wedge e(y_1,y_2) \wedge \cdots \wedge e(y_{m-1},y_m))]].$$

By the definition of K^*_{-1}, this gives a proof of

$$\begin{aligned}(\forall x)(\exists q)(\exists u)(\exists v)(\exists y_m)\cdots(\exists y_1)& \\ [G \supset (ID(y_m,q,u,v) \wedge qDfnt(q) \wedge & \\ e(x,y_1) \wedge e(y_1,y_2) \wedge \cdots \wedge e(y_{m-1},y_m))]&.\end{aligned} \tag{2}$$

Let $\phi_{m,i}$ be the sentence (note that x and q have been replaced by 0 and q_i)

$$(\exists u)(\exists v)(\exists y_m)\cdots(\exists y_1) [G \supset (ID(y_m,q_i,u,v) \wedge e(0,y_1) \wedge e(y_1,y_2) \wedge \cdots \wedge e(y_{m-1},y_m))].$$

Then, from equation (2) and the definition of *qDfnt*, there is a polynomial size proof of

$$\bigvee_{i=1}^{s} \phi_{m,i}. \tag{3}$$

Considering again the standard, classical model with domain the set of integers and with the non-logical symbols having their intended interpretations, it is clear that $\phi_{m,i}$ is a true sentence if and only if Turing machine M is in state q_i at time $2 \Uparrow m$. Since equation (3) has an intuitionistic proof of size $n = m^{O(1)}$, and by the choice of M, we have proved the desired superexponential lower bound on the complexity of deciding the disjunction property.

Acknowledgement. We thank for P. Pudlák for the suggesting the investigation of the disjunction property of propositional logic to us. J. Johannsen helped us with comments and corrections to an earlier version of this paper.

References

[1] G. Gentzen, *Untersuchungen über das logische Schliessen*, Mathematische Zeitschrift, 39 (1934), pp. 176–210, 405–431. English translation in [2], pp. 68-131.

[2] ——, *Collected Papers of Gerhard Gentzen*, North-Holland, 1969. Edited by M. E. Szabo.

[3] R. E. Ladner, *The computational complexity of provability in systems of modal propositional logic*, SIAM Journal on Computing, 6 (1977), pp. 467–480.

[4] V. P. Orevkov, *Lower bounds for lengthening of proofs after cut-elimination*, Zapiski Nauchnykh Seminarov LOMI, 88 (1979), pp. 137–162. In Russian: English translation: *J. Soviet Mathematics* 20 (1982) 2337-2350.

[5] D. Prawitz, *Natural Deduction: A Proof-Theoretical Study*, Almqvist & Wiksell, Stockholm, 1965.

[6] P. Pudlák, *The lengths of proofs*, in Handbook of Proof Theory, S. R. Buss, ed., Elsevier North-Holland, 1998, pp. 547–637.

[7] R. Statman, *Intuitionistic propositional logic is polynomial-space complete*, Theoretical Computer Science, 9 (1979), pp. 67–72.

[8] ——, *Lower bound on Herbrand's theorem*, Proceedings of the American Mathematical Society, 75 (1979), pp. 104–107.