A Complexity gap for tree-resolution

Søren Riis *

June 2001

Abstract

This paper shows that any sequence ψ_n of tautologies which expresses the validity of a fixed combinatorial principle either is "easy" i.e. has polynomial size tree-resolution proofs or is "difficult" i.e requires exponential size tree-resolution proofs. It is shown that the class of tautologies which are hard (for tree-resolution) is identical to the class of tautologies which are based on combinatorial principles which are violated for infinite sets. Further it is shown that the gap-phenomena is valid for tautologies based on infinite mathematical theories (i.e. not just based on a single proposition).

A corollary to this classification is that it is undecidable whether a sequence ψ_n has polynomial size tree-resolution proofs or requires exponential size tree-resolution proofs. It also follows that the degree of the polynomial in the polynomial size (in case it exists) is non-recursive, but semi-decidable.

Keywords: Logical aspects of Complexity, Propositional proof complexity, Resolution proofs.

1 General motivation

In this paper a new kind of result for propositional logic is introduced. It is shown for a large class of uniform families of unsatisfiability problems $\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_j, \ldots$ that the family either has polynomial size tree-resolution refutations or requires full exponential size tree-resolution refutations. Thus intermediate growth rates like for example $2^{\log^k(n)}, k = 2, 3, \ldots$ never occur. For non-uniform families (where, for example, each $\mathcal{C}_j$ might express a

[1] Department of Computer Science, Queen Mary, University of London London E1 4NS, United Kingdom. Email: smriis@dcs.qmw.ac.uk Phone: +44 020-7882 7611

different combinatorial principle) there is no complexity gap and any super-polynomial but sub-exponential growth-rate can appear. Somewhat informally the main result states that if the sequence C_j expresses the *same* combinatorial principle for each j, then there is a complexity gap for tree-resolution.

In complexity theory we are particularly interested in what happens to a set of decision problems that are *similar* except for size. The main concern is with what happens to the computational complexity as the size of the problem tends to infinity. In propositional complexity we frequently study what happens to a collection of tautologies which are *similar* except for size. Here the main concern is what happens to the proof complexity (for a given proof system) as the size of the tautologies tends to infinity.

One interesting general feature in Complexity Theory is that while a few complexities seem to appear again and again, other complexities virtually never appear. A somewhat similar point is made by Hardy in [19] where he observes:

> No function has yet presented itself in analysis the laws of whose increase, in so far as they can be stated at all, cannot be stated, so to say, in logarithmic-exponential terms.

If Hardy's remark is valid for complexity theory and proof complexity in general we will expect - due to the discrete nature of the logarithmic-exponential growth rates - that complexity gaps are widespread and part of a general phenomenon.

An important motivation for studying propositional proof systems is tied up with the following basic question: Given a true statement (tautology) what is the length of the shortest proof of the statement? Here the answer depends, of course, on which axiomatic proof system is being used. From a computer science perspective the question is particularly fundamental for propositional logic. A vast number of problems that occur in computer science including knowledge-representation, learning, planning, automated theorem proving, verification etc. are linked to this problem. As formalised by Cook and Reckhow [14], there exists a propositional proof system in which any tautology ψ has a proof of size bounded by $p(|\psi|)$ for a fixed polynomial p if and only if NP=co-NP. This question is far beyond current techniques. However, Cook and Reckhow proposed a program of research which systematically tries to obtain non-polynomial lower bounds for stronger and stronger propositional proof systems. The hope is that this will eventually lead to a separation of NP from co-NP.

Tautologies expressing versions of the pigeonhole principle (and other related combinatorial principles) have played an important role in obtaining lower bounds for the length of propositional proof. The same principles have been widely used to separate propositional proof systems.

The first super-polynomial lower bound for resolution (satisfying a restriction called regularity) was obtained by Tseitin [38]. Subsequent work simplified Tseitin's proof and improved the lower bounds for regular resolution [16], [39]. However great difficulty was experienced in extending Tseitin's arguments to unrestricted resolution (=dag-resolution). The hurdle was first overcome when Haken managed to give a super-polynomial lower bound for the pigeonhole principle for dag-resolution [18]. This result was later improved considerably by Ajtai [1], [2] to a super-polynomial lower bound on bounded depth Frege proofs. Ajtai also used his approach to show independence results from Bounded Arithmetic. These results were subsequently improved in various ways and generalities [3], [4], [7], [31], [32].

In this paper we consider a quite general class of tautologies. The tautologies like the pigeonhole principle (or the related Counting modulo q principles) studied in [1], [2], [3], [4], [7], [18], [31], [32] can be viewed as a special case of a translation where a given universal second order sentence is translated into a sequence of tautologies in propositional logic. A sequence of tautologies for the pigeonhole principle appears for example by translating

$$\forall f \forall c\ (\exists x\ f(x) = c\ \vee\ \exists x, y\ x \neq y\ \&\ f(x) = f(y))$$

into propositional logic. More specifically for each n we can translate the fact that the first order sentence $\forall x\ f(x) \neq c\ \wedge\ (\forall x, y\ x = y \vee f(x) \neq f(y))$ has no models for size n into an unsatisfiable system of clauses. This translation technique is very general and allows us make formal sense of what it means for a class of tautologies to be similar except for size [34].

In section 3 we review this translation procedure in more details. In that section we also state the main theorem and discuss some corollaries. The proof of the main theorem consists of two parts - the lower bound and the upper bound. The lower bound is exponential, while the upper bound is polynomial. This is no contradiction since the bounds apply to different situations.

In the section *further perspectives* it is shown that it is possible to assign a mathematical theory $T_P(C)$ to any given propositional proof system P and any proof size complexity $C = C(n)$ (like e.g. *polynomial size* $n^{O(1)}$, *size* $n^{\log^{O(1)}(n)}$ *or size* $2^{O(n)}$*).* This idea is new and places the main result in a larger perspective. For any propositional proof system P one can ask about the behaviour of $T_P(C)$ when the proof size resources C increase. This question is well defined, and is linked to the complexity gap phenomena I introduce in this paper. The paper raises a number of questions related to the theory T_P. I conjecture that many other propositioanal proof systems also have a complexity gap.

2 Tree resolution

In this section I give a brief reminder of some of the basic concepts related to resolution proofs.

A *literal* is a propositional variable or the negation of a propositional variable. A clause $C := \{l_1, l_2, \ldots, l_u\}$ is a collection of literals, and it is satisfied exactly when the disjunction $l_1 \vee l_2 \vee \ldots \vee l_u$ holds. In the famous NP-complete problem 3-SAT, the decision problem is to decide if a given collection of clauses (which each contain at most 3 literals) is satisfiable.

Resolution is a refutation system designed to provide certificates (i.e. proofs) that a system of clauses is unsatisfiable. A given formula is shown to be a tautology by showing that its negation, put into conjunctive normal form (i.e. clausal form) is unsatisfiable. This is done by means of the resolution rule

$$\textit{Resolution rule}: \qquad \frac{C_1 \cup \{p\} \quad C_2 \cup \{\neg p\}}{C_1 \cup C_2}$$

The given clauses are often referred to as *axioms*, and the task is to derive the empty clause (the contradiction) from the axioms. In *tree-resolution* the proof is organised as a binary tree with the axioms in the leaves and the empty clause in the root. In *dag-resolution* (or just *resolution*) the proof is given as a linear sequence $C_1, C_2, \ldots, C_u$ of clauses, where each clause either is an axiom or can be obtained by means of the resolution rule (applied to two already derived clauses). In a resolution proof, clauses can be reused more the once. A tree-resolution proof do not allow this. In this paper we will only consider tree-resolution proofs.

3 Translating logic into propositional logic

Informally the first part of the result can be stated as follows: Let ψ_n be a sequence of tautologies which for each n expresses the validity of a fixed combinatorial principle $\mathcal{P}_{\text{com}}$. The main result states that for any such sequence ψ_n either the sequence has polynomial size tree-resolution proofs or the sequence requires truly exponentially size tree-resolution proofs.

We have of course to make precise what it is for a sequence ψ_n of tautologies to express a combinatorial principle. The approach I take is to consider combinatorial principles to be statements which holds for all structures (including graphs and hyper-graphs) on n vertex. Since universal second order properties on graphs (hyper-graphs) defines co-NP, universal second order properties is clearly the largest class of graph-properties which we meaningfully will expect has straight forward translations into propositional logic (since the set of tautologies is co-NP complete).

This however is the same class a the set of first order formulas which is valid in all models of size n. Thus we assume the $\mathcal{P}_{\text{com}}$ is of the form: "ψ has no models of size n" for some first order sentence ψ.

I would like to describe the procedure for translating such combinatorial principles into sequences of sets of clauses in propositional logic.

Instead of describing the translation method in full detail, a single example will be used - the theory DLO of dense order without endpoints (this example is due to Urquhart [41]). Readers who want to see the description of the translation procedure in full generality are refered to [34]. The theory DLO has a single binary relation $\prec$ as it only a primitive non-logical symbol. Its non-logical axioms are as follows:

(1) $(x \prec y \ \wedge \ y \prec z) \rightarrow x \prec z$

(2) $x \prec y \rightarrow \exists z(x \prec z \ \wedge \ z \prec y)$

(3) $x \prec y \vee x = y \vee y \prec x$

(4) $\neg(x \prec x)$

(5) $\exists y \quad (x \prec y)$

(6) $\exists y \quad (y \prec x)$

The first step is to "skolemize", i.e. to convert the above theory to an equivalent purely universal theory by introducing Skolem functions to replace existential quantifiers (see for example Chapter 3 in Hodges's textbook [20]). In the case of DLO, we add three new function symbols, f, g, h to the underlying language. Further, the second, fifth and sixth axioms are written as universal axioms:

(2a) $x \prec y \rightarrow (x \prec f(x,y) \ \wedge \ f(x,y) \prec y)$

(5a) $x \prec g(x)$

(6a) $h(x) \prec x.$

The next step is to rewrite the theory so that there are no embedded function symbols, so that all atomic formulas in the axioms are of the form $R(\vec{x}), f(\vec{y}) = z$ or $x = y$, where $\vec{x}$ and $\vec{y}$ are sequences of variables. An atomic formula of this form will be decribed as *basic*. Constants are treated as zero-place function symbols, so that, for example, the formula $0 < 1$ is not a basic formula, although $x < 1$ and $0 < x$ are basic. A formula of the form $x = f(\vec{y})$ is not viewed as a basic formula.

In our example DLO, the only axioms involving embedded function symbols are the new axioms **2a**, **5a** and **6a**. We replace these by:

(**2b**) $(x \prec y \ \wedge \ f(x, y) = z) \rightarrow (x \prec z \ \wedge \ z \prec y)$

(**5b**) $g(x) = y \rightarrow x \prec y$

(**6b**) $h(x) = y \rightarrow y \prec x$

Finally, we rewrite the axioms of the new purely universal theory as conjunctions of clauses. In the case of the theory derived from DLO, there is almost nothing to do, since all of the axioms are Horn formulas except for the (new) axiom **2b**.

Let T be the theory that results from the preceding sequence of transformations. It is important to note that the original theory and the transformed theory T have essentially the same set of models; to be more precise, every model of the original theory can be expanded to a model of the transformed theory, and every model of the transformed theory can be cut down to a model of the original theory.

We now describe the definition of the sequence of contradictory sets of clauses derived from the theory T. We form the set $\mathcal{S}_{T,n}$ of clauses that express the that there is a model of T of size n. Let $\mathrm{Var}_n := \{c_1, c_2, \ldots, c_n\}$ be new constants not appearing in the theory. We form the set of clauses $\mathcal{C}_n$ that results from replacing all variables in an axiom of T by constants from Var_n in all possible ways.

Let us regard the atomic sentences in $\mathcal{S}_{T,n}$ as propositional letters (that is to say, we disregard their internal structure). Each atomic sentence $c_i = c_j$ is replaced by true/false depending on whether $i = j$ or $i \neq j$. This ensures that the constants $c_1, c_2, \ldots, c_n$ are distinct and that there are at least n things in the universe. In order to distinguish between an atomic sentence ψ and the propositional letter corresponding to it, we shall enclose ψ in corner quotes, so that $\ulcorner\psi\urcorner$ is the propositional letter corresponding to ψ. We shall say that an atomic sentence is *basic* if it results from a basic formula by substituting constants from Var_n for all the variables. The theory DLO leads to the system $\mathcal{S}_{\mathrm{DLO},n}$ of propositional logic which includes the clausal form of the propositions:

(1)$'$ $r_{ij} \wedge r_{jk} \rightarrow r_{ik}, \qquad i, j \in \{1, 2, \ldots, n\}$

(2b)$'$ $(r_{ij} \wedge f_{ijk}) \rightarrow (r_{ik} \wedge r_{kj}), \qquad i, j, k \in \{1, 2, \ldots, n\}$

(3)$'$ $r_{ij} \vee r_{ji}, \qquad i \neq j, i, j \in \{1, 2, \ldots, n\}$

(4)$'$ $\neg r_{ii}, \qquad i \in \{1, 2, \ldots, n\}$

(5b)$'$ $g_{ij} \rightarrow r_{ij}, \qquad i, j \in \{1, 2, \ldots, n\}$

(6b)$'$ $h_{ij} \rightarrow r_{ji}, \qquad i, j \in \{1, 2, \ldots, n\}$

where r_{ij} be shorthand for $\lceil c_i \prec c_j \rceil$, f_{ijk} shorthand for $\lceil f(c_i, c_j) = c_k \rceil$, g_{ij} shorthand for $\lceil g(c_i) = c_j \rceil$ and h_{ij} shorthand for $\lceil h(c_i) = c_j \rceil$.

Besides these clauses $\mathcal{S}_{T,n}$ also consist of a set of clauses $\Gamma_{\leq n}$ expressing that there are at most n things in the universe. More specifically $\Gamma_{\leq n}$ consists of all clauses obtained by substituting a term of the form $f(\vec{a})$ in the formula $(x = c_1 \vee x = c_2 \vee \ldots \vee x = c_n)$, where $\vec{a}$ is a sequence of constants from Var_n. Finally for each function symbol f we include the clauses $\{\neg\lceil f(\vec{a}) = c_j \rceil, \neg\lceil f(\vec{a}) = c_m \rceil\}$ where $j, m \in \{1, 2, \ldots, n\}$ are distinct elements. In our example we get:

(i) $f_{ij1} \vee f_{ij2} \vee \ldots \vee f_{ijn}, \qquad i, j \in \{1, 2, \ldots, n\}$

(ii) $g_{i1} \vee g_{i2} \vee \ldots \vee g_{in}, \qquad i \in \{1, 2, \ldots, n\}$

(iii) $h_{i1} \vee h_{i2} \vee \ldots \vee h_{in}, \qquad i \in \{1, 2, \ldots, n\}$

(iv) $\neg f_{ijk} \vee \neg f_{ijl}, \qquad i, j, k \neq l \in \{1, 2, \ldots, n\}$

(v) $\neg g_{ij} \vee \neg g_{ik}, \qquad i, j \neq k \in \{1, 2, \ldots, n\}$

(vi) $\neg h_{ij} \vee \neg h_{ik}, \qquad i, j \neq k \in \{1, 2, \ldots, n\}$

This completes the translation of the theory DLO. The propositional translation of DLO consists of $\mathcal{S}_{T,n}$ which precisely consists of clausal versions of $(1)', (2b)', (3)', (4), (5b)', (6b)'$ as well as (i)-(vi). As the theory DLO has no finite models (in particular no models of size n) $\mathcal{S}_{\mathrm{DLO},n}$ is unsatisfiable. In general:

Lemma 3.1 *([41]) If T is a first order theory, and $\mathcal{S}_{T,n}$ a set of clauses derived from T by the above construction, then $\mathcal{S}_{T,n}$ is satisfiable (in the sense of propositional logic) if and only if T has a model of size n.*

Furthermore if T consists of the conjunction of finitely many first order sentences, then there exists a constant $c > 0$ such that $\mathcal{S}_{T,n}$ contains less than n^c symbols for each $n \geq 2$.

Proof: If $\mathcal{S}_{T,n}$ is satisfied by an interpretation I, then we can construct a model M on the universe $\{1,2,\ldots,n\}$ by interpreting the constant c_i as standing for the integer i, and interpreting the relation and function symbols in accordance with which atomic formulas are true in I. That is to say, if f is a function symbol in the language of T, and $\vec{c}$ a sequence of constants in Var_n, then we make $f(\vec{c}) = c_i$ true in the model M if and only if $\lceil f(\vec{c}) = c_i \rceil$ is true in the interpretation I. If R is a relation symbol in the language of T, then we make $R(\vec{c})$ true in M if and only if $\lceil R(\vec{c}) \rceil$ is true in I. Since we have included the axiom $(f(\vec{x}) = y \ \wedge \ f(\vec{x}) = z) \rightarrow y = z$ in our theory, it follows that exactly one propositional letter of the form $\lceil f(\vec{c}) = c_i \rceil$ is true in I, since $\mathcal{S}_{T,n}$ includes $\mathcal{C}_n$. Hence, the functions in M are well defined.

We need to verify that all axioms of T are true in M. Since the axioms involves only basic atomic formulas, it is sufficient to show that if $\psi(\vec{x})$ is a basic atomic formula, and $\vec{a}$ a sequence of integers in $\{1,2,\ldots,n\}$, then $\psi(\vec{a})$ is true in M if and only if $\lceil \psi(c(\vec{a})) \rceil$ is true in I, where $c(\vec{a})$ is a sequence of constants from Var_n in which the successive subscripts form the vector $\vec{a}$. For basic formulas of the form holds true by definition of M. The only remaining type of basic formula is $x = y$; the claim holds in this case by the axiom $x = x$ and the fact that $\mathcal{S}_{T,n}$ includes all of the sentences $\neg(c_i = c_j)$ for $i \neq j$.

If the theory from which we begin is (as in the case of our example DLO) a finitely axiomatisable theory, then the procedure for producing the set of clauses $\mathcal{S}_{T,n}$ can be expressed as a polynomial time algorithm. The size of $\mathcal{S}_{T,n}$ is bound by a polynomial in n. □

As stated in lemma 3.1, if T has no finite models (as is the case for our example DLO), all of the sets of clauses in the sequence $\mathcal{S}_{T,n}, n = 1,2,\ldots,$ are contradictory, and hence have tree resolution refutations. Our main theorem concerns such refutations.

Before we proceed it is important to notice that the above *formal translation* of any given theory T (into clauses $\mathcal{S}_{T,n}$) is highly natural and is in agreement (modulo minor syntactic changes) with the informal but canonical result one would get without knowing the above procedure. The translation procedure results in a system of clauses (like $(1)', (2b)', (3)', (4), (5b)', (6b)'$ in our example) which is a syntactial reformulation of the skolemised theory T. Besides this for each function symbol the translation includes clauses (like (i)-(vi)) which ensure that functions and constants are well behaved (i.e. have one and only one value).

The translation corresponds (except from a minor difference in the treatment of constants in the original language for T) to the informal procedure which seems to have been used when considering a principle like the pigeonhole principle [14] or the parity principle [3]. The translation also agrees with the procedure defined in [34]. Thus it is no coincidence that the propositional version of the pigeonhole principle first studied by Haken [18], is essentially the same version one gets by translating the pigeonhole principle (as stated in predicate logic) as described above. The same holds for many other combinatorial principles already studied in the literature.

Notice however that the translation is not is unique. There is usually more than one way to write a propositional formula as a conjuction of clauses. And more significantly there is usually more than one way of skolemising a theory T. In section 6 we will make a slight extension in the set of translation procedures such that any reasonable translation procedure is covered. In section 6 we show that our main result is robust and remain valid for all the translation procedures we consider.

The alert reader will have noticed that the system of clauses $\mathcal{S}_{T,n}$ is closed under the natural action of the symmetrical group S_n. This is not surprising as the propositional version of the sentence "T has a model of size n" is independent of the underlying interpretation of the constants $c_1, c_2, \ldots, c_n$. It turns out that the class of tautologies we can get from translating propositions in logic in propositional logic is exactly the class of S_n-generated tautologies we defined in [34]. The S_n-symmetry plays a crucial role especially when considering algebraic proof complexity and link these problems to representation theory - a central and well studied area in mathematics (see [35] for more details).

For a theory T we let k_T^{rel} denote the maximal arity of a relation symbol in the language of T, and let k_T^{fun} denote the maximal arity of a function symbol in the language of T. We can now state our result as follows:

Theorem 3.2 *Let T be a first order theory (which might not be finitely axiomatisable and which might be highly non-recursive). Let $\mathcal{S}_{T,n}$ denote the satisfiability problem which results from applying the natural translation procedure which translates the statement "T has a model of size n" into propositional logic. There are two possibilities:*

(1) For each value of n for which $\mathcal{S}_{T,n}$ is unsatisfiable, the smallest tree-resolution refutations have size at least $2^{n/\max(k_T^{\text{rel}},1+k_T^{\text{fun}})}$

(2) Asymptotically (i.e when n tends to infinity) $\mathcal{S}_{T,n}$ has polynomial size (in n) tree-resolution refutations.

Possibility (1) happens if and only if T has an infinite model. The lower bound in (1) holds whenever $\mathcal{S}_{T,n'}$ is satisfiable for some $n' > n$.

The theorem gives a complete classification of the theories T for which $\mathcal{S}_{T,n}$ requires large tree-resolution refutations (if there are any at all - in general $\mathcal{S}_{T,n}$ could be satisfiable). More specifically, a theory T leads to hard (for tree-resolution) tautologies if and only if T has an infinite model.

Notice that the result is expressed in terms of n - the size of the model - and not the size of the satisfiability problem (which is either infinite with infinitely many variables - when T is infinite - or has size n^c for some constant $c > 0$).

Corollary 3.3 *There is no decision procedure which given a first order formula ψ as input, decides whether the sequence $\mathcal{S}_{\psi,n}$ has polynomial size tree-refutations or requires exponential size tree-refutations. More specifically the collection of ψ which have polynomial size tree-refutations is non-recursive (but recursively enumerable).*

Proof: The collection A of first order sentences which has an infinite model is - according to Trachtenbrots theorem - non-recursive. The collection A has a complement which is recursively enumerable. □

Theorem 3.2 is optimal in the following sense:

Proposition 3.4 *There exists ψ for which the sequence $\mathcal{S}_{\psi,n}$ has optimal tree-resolution refutations of size $2^{\Omega(n)}$.*

Proof: This proposition is non-trivial since n denotes the size of the model rather than the number of variables. Consider $\mathcal{S}_{\mathrm{DLO},n}$. The lower bound of $2^{\Omega(n)}$ follows from Theorem 3.2. The upper bound of $2^{O(n)}$ follows by induction on n. This argument is especially straightforward if we view tree-resolution refutations as decision trees (see next section for more details). □

The polynomial size upper bound in Theorem 3.2 cannot be improved by choosing the polynomial degree so it depends only on simple syntactic properties of ψ.

Theorem 3.5 *There is no total recursive function $r = r(`\psi')$ which given a first order sentence ψ as input, produce output $r \in N$, such that the sequence $\mathcal{S}_{\psi,n}$ is guaranteed to have either $\leq n^r$-size tree-refutations (for n sufficiently large) or to require exponential size tree-refutations.*

This theorem shows that the exponent in the polynomial bound can be arbitrarily bad. More specifically:

Corollary 3.6 *For any total recursive function F however fast-growing, e.g. the Ackerman function, $F_{\epsilon_0}, F_{\Gamma_0}$ (see for example [37] for a survey of fast growing functions), there exists a sentence ψ such that the sequence $\mathcal{S}_{\psi,n}$ has polynomial size refutations. But the degree of the polynomial needed to bound the size of the smallest tree-refutations, is larger than $F(|\psi|)$ where $|\psi|$ denotes the number of symbols in ψ.*

Let me emphasise that the method of generating uniform families of tautologies we discussed above is quite powerful. Actually (assuming NEXP $\neq$ co-NEXP) the method of getting tautologies from translating statements in predicate logic is rich enough to generate a universally difficult sequence $\mathcal{S}_{\psi,n_1}, \mathcal{S}_{\psi,n_2}, \ldots$ of unsatisfiable collections of clauses which require non-polynomial size refutations for any given propositional proof system [34]. Notice that main result (Theorem 3.2) includes tautologies of this general form (where $\mathcal{S}_{\psi,n}$ only is unsatisfiable for $n \in A$ for some infinite subset $A \subseteq N$).

The philosophy behind Theorem 3.2 was first articulated in [29] where it was shown (in the context of Bounded Arithmetic) that combinatorial principles which fail as infinitary combinatorics in a sense (which can be made precise) are harder (to prove) than combinatorial principles which are also valid as part of infinitary combinatorics. More specifically, in [29] I showed that combinatorial principles which fail for infinite sets can never be proven on the first tree levels $S_2^1(\alpha) \subseteq T_2^1(\alpha) \subseteq S_2^2(\alpha)$ of Sam Buss hierarchy of Bounded Arithmetic, while such combinatorial principles in certain cases can be proven on the fourth level $T_2^2(\alpha)$. It is well known that provability in fragments of Bounded Arithmetic is closely related to propositional proof complexity (for more details see [21]). The results in the present paper are, however, technically unrelated to the results in [29]. The proof technique in the current paper is different from the forcing technique which was employed in [29]. Jan Krajicek has pointed out (personal communication) that the exponential lower bound in Theorem 3.2 follows by a modification of his proof of Theorem 11.3.2 in [21] (which is essentially the main result in [29]). See also Lemma 9.5.2 in [21] where this is stated explicitly.

I am aware of only one other result which gives a complexity gap between polynomial complexity and exponential complexity. A beautiful result [17] relates the Vapnik-Chervonenkis (VC) dimension to the growth rate of the complexity of learning the concept class C. It states that this growth rate is either polynomial or exponential. Furthermore, it is polynomial if and only if the VC-dimension of C is finite. The underlying mathematics in this result is completely different from the one behind the complexity gap theorem

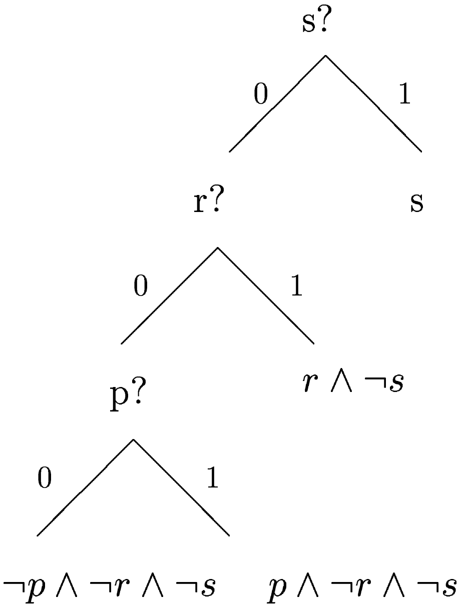
(Theorem 3.2). It is however remarkable that the dichotomy of finite versus infinite plays a crucial role in both the VC-complexity gap theorem as well as in the complexity gap theorem.

4 The exponential size lower bound

For the lower bound it is convenient to view a tree-resolution refutation as a decision tree. This is essentially done by turning the refutation tree on its head. On an input (i.e. a truth assignment) the decision tree outputs an unsatisfied clause. To illustrate the idea, consider for example, the tree-refutation:

$$\dfrac{\dfrac{\dfrac{\{p,r\} \quad \{\bar{p},s\}}{\{r,s\}} \quad \{\bar{r},s\}}{\{s\}} \quad \{\bar{s}\}}{\emptyset}$$

The refutation shows that $\{\{\bar{s}\},\{p,r\},\{\bar{p},s\},\{\bar{r},s\}\}$ is unsatisfiable. If we turn the refutation on its head we get the binary decision tree.


s?
0
1
r?
s
0
1
p?
$r \wedge \neg s$
0
1
$\neg p \wedge \neg r \wedge \neg s$
$p \wedge \neg r \wedge \neg s$

Decision tree refuting $\{\{\neg s\}$,$\{\neg r, s\}$,$\{\neg p, s\}$, {p , r}}

For any truth assignment of the variables (s, p and r) the decision tree uniquely determines a clause which is unsatisfiable. Notice that each branch leads to a conjunction which violates a clause.

In order to simplify the situation further I introduce an idea inspired by [28]. The idea is to define and analyse a game between a *prover* and an *adversary.* The prover (female) holds a decision tree while the task of the adversary (male) is to make up answers to the questions put forward by the prover. The adversary claims (mistakenly) that a set S of clauses is satisfiable. The prover wins if the adversary is caught in an elementary lie (contradiction). An elementary lie appears if the adversary has produced answers which violate a clause in S. A strategy for the prover is a decision tree. The answers of the adversary determine a path through the decision tree. Each leaf in the decision trees corresponds to an elementary contradiction. The task of the adversary is to survive as many questions as possible. We observe that the adversary can survive no more than $h - 1$ questions where h is the length of the longest branch through the decision tree.

We want to produce a lower bound on the size of any decision tree (rather than on the longest branch) so we modify the rules of the game slightly. To do this we change the task of the adversary. The (modified) task of the adversary is to embarrass the prover as much as possible with out being caught in a lie. An embarrassment appears whenever the adversary gives the prover *a free choice* . In that case the prover is allowed to answer the question the way she prefers.

To be slightly more formal for any given system $\mathcal{S}$ of clauses and for any $k \in N$ we define the game $G(\mathcal{S}, k)$ as follows: The prover asks questions (concerning the truth value of propositional variables). Given a question the adversary has two options: He can either answer the question or give the prover a free choice. In the latter case the prover has to choose an answer to the question (this answer should of course be known to the adversary). The adversary scores one point every time a free choice is given to the prover. The adversary wins if no elementary contradiction is reached before he has scored k points. To avoid the game dragging on indefinitely (by for example the prover keeps asking the same question), we design the rules such that the adversary wins if the prover ask the same question twice.

Lemma 4.1 *If the adversary has a winning strategy in the game $G(\mathcal{S}, k)$ each tree resolution refutation of $\mathcal{S}$ contains the binary subtree which has height k and size $2^{k+1} - 1$. Consequently each tree resolution refutation of $\mathcal{S}$ must have size $\geq 2^{k+1} - 1$.*

Proof: Assume the adversary has a winning strategy in the game $G(\mathcal{S}, k)$. Let $\mathcal{T}$ be an arbitrary decision tree for $\mathcal{S}$. We can assume the same question is only asked once along each branch in $\mathcal{T}$. It suffices to show that $\mathcal{T}$ must contain a binary subtree of hight k and size $2^{k+1} - 1$. The strategy of the adversary determines a subtree $\mathcal{T}'$ which consists of all possible plays which can appear when the adversery sticks to his strategy. Each node in $\mathcal{T}'$ has either one or two children. A node has one child whenever the question corresponding to the node is answered. A node has two children whenever the question corresponding to the node was left as a free choice. The adversary has a winning strategy in $G(\mathcal{S}, k)$ so each branch in $\mathcal{T}'$ has at least k branching points. Thus $\mathcal{T}'$ (which is a subtree of $\mathcal{T}$) contains the binary subtree which has height k and size $2^{k+1} - 1$. □

Next assume that we are given a first order theory T as well as $n \in N$. Consider $\mathcal{S}_{T,n}$.

Lemma 4.2 *Assume there is a model M of T which contains more than n elements. Then the adversary has a winning strategy in the game* $G(\mathcal{S}_{T,n}, (n - k_T^{\text{fun}})/\max(k_T^{\text{rel}}, 1 + k_T^{\text{fun}}))$.

Proof: Assume that there is a model M of T which contains more than n elements (possibly infinitely many). Using the existence of such a model we devise a winning strategy for the adversary. Essentially the adversary claims that there is model $\tilde{M}$ which has n elements. Consider any strategy for the prover and let τ denote the decision tree corresponding to this strategy. The prover asks questions about this hypothetical model $\tilde{M}$. Let $\tilde{U}$ denote the class of models of M which contains at least n points and which has the constants $c_1, c_2, \ldots, c_n$ interpreted as distinct elements. We assumed that there is a model M of T which contains more than n elements so $\tilde{U}$ is non-empty. The adversery's strategy involves many models in $\tilde{U}$ rather than a single fixed model.

Consider a node v in the decision tree τ. The decisions made until that node is a conjunction of basic atomic sentences and negation of basic atomic sentences. Let Ψ_v denote this conjunction. Informally Ψ_v represent provers knowledge of the model $\tilde{M}$ adversary claims exists.

Let $\lceil R(s_1, s_2, \ldots, s_r) \rceil$, $s_1, s_2, \ldots, s_r \in \{c_1, c_2, \ldots, c_n\}$ denote the question assigned to the node v. If all models $M \in \tilde{U}$ which satisfies Ψ_v satisfies $R(s_1, s_2, \ldots, s_r)$ the adversary answers "yes". If all models $M \in \tilde{U}$ which satisfies Ψ satisfies $\neg R(s_1, s_2, \ldots, s_r)$ the adversary answers "no". If there exists (at least) two models $M_1, M_2 \in \tilde{U}$ which satisfies Ψ_v, such that M_1 satisfies $R(s_1, s_2, \ldots, s_r)$ while M_2 satisfies $\neg R(s_1, s_2, \ldots, s_r)$ the adversary gives the prover a free choice. We then refer to the node as a 'free choice' node. To show the lemma it is sufficient to show:

Claim A: *This strategy guarantee the adversary a win (i.e. the adversery can always produce at least $(n - k_T^{\rm fun})/\max(k_T^{\rm rel}, 1 + k_T^{\rm fun}))$ free choices in an actual game)*

proof of claim A: Assume that the prover can win against this strategy i.e. has a decision tree $\mathcal{T}$ which guarantees her a win. As before let $\mathcal{T}'$ denote the subtree of all possible plays (when adversary follows the strategy defined above). Fix an arbitrary branch $\mathbf{b}$ in $\mathcal{T}'$ (i.e. a possible play which can appear when the adversary follows the strategy defined above). Consider the collection of constants $c_1, c_2, \ldots, c_n$ which appears in the basic atomic sentences which appears in 'free choice' points along $\mathbf{b}$. Notice that each 'free choice' point corresponds to a branching point in $\mathcal{T}'$. If there are k 'free choice' point along the branch there is at most $k \times \max(k_T^{\rm rel}, 1 + u)$ such constants (where $u := k_T^{\rm fun}$). It suffices to show that each of the constants $c_1, c_2, \ldots, c_n$ (with at most u exceptions) must appear in at least once in a basic atomic formula associated to a 'free choice' point along $\mathbf{b}$. Before completing the proof of the claim we show:
Claim B: *At least $n - u$ of the constants $c_1, c_2, \ldots, c_n$ appear in a basic atomic formula associated to a 'free choice' node* $\mathbf{b}$.

Proof of claim B: Assume that none of the constants $c_{i_1}, c_{i_2}, \ldots, c_{i_u}, c_{i_{u+1}}$ appear in a basic atomic formula associated to a 'free choice' node along some branch $\mathbf{b}$ in τ. To show the subclaim it suffices to show this leads to a contradiction. We consider a branch $\mathbf{b}$ through $\mathcal{T}'$ so there must be at least one model $M \in \tilde{U}$ which satisfies all decisions along the branch. Since the decision tree lead to an elementary contradiction it must be because one of the clauses in $\mathcal{S}_{T,n}$ is unsatisfied. This unsatisfied clause much be a clause which is a weakening of a clause of the form $\{f(\vec{s}) = c_1, f(\vec{s}) = c_2, \ldots, f(\vec{s}) = c_n\}$ since all other types of clauses are satisfied by any model $M \in \tilde{U}$. Thus the branch $\mathbf{b}$ must involve the conjunction A of decisions $f(\vec{s}) \neq c_1, f(\vec{s}) \neq c_2, \ldots, f(\vec{s}) \neq c_n$ as well as possible some other decisions (here $\vec{s}$ denote a $\leq u$-tuple of constants $s_1, s_2, \ldots, s_u \in \{c_1, c_2, \ldots, c_n\}$). According to the pigeonhole principle there must be at least one constant $c' \in \{c_{i_1}, c_{i_2}, \ldots, c_{i_u}, c_{i_{u+1}}\}$ such that $c' \notin \{s_1, s_2, \ldots, s_u\}$. Since none of the constants $\{c_{i_1}, c_{i_2}, \ldots, c_{i_u}, c_{i_{u+1}}\}$ appear in an atomic formula associated to 'free choice' node along $\mathbf{b}$, c' does not appear in any atomic formula associated to 'free choice' node along $\mathbf{b}$. Thus the decision $f(\vec{s}) \neq c'$ was not made as a free choice and thus must have been a forced answer. Thus all models $M \in \tilde{U}$ which satisfies the conjunction A must have $f(\vec{s}) \neq c'$. This leads to a contradiction. To see this let $M \in \tilde{U}$ be a model which satisfies the conjunction A. Consider the model M' which is identical to M except c' is interpreted such that $f(\vec{s}) = c'$ (since c' does not appears in $\vec{s}$ there is such a model). We claim that:

Claim C: *M' satisfies the conjunction A.*

Proof of claim C: If M' does not satisfy the conjunction A there exists a (negation of) basic atomic sentence $R(\vec{c})$ which is in the conjunct A, but which is not satisfied by M'. Consider the first node w along the branch $\mathbf{b}$ where (the negation of) the basic atomic sentence in the conjunction A not is satisfied by M'. Since M and M' are identical except for the intepretation of c' the sentence $R(\vec{c})$ must involve the constant c'. Let me explain this point in details. For each vector $\vec{c}$ of constants from $\{c_1, c_2, \dots, c_n\} \setminus \{c'\}$, the model M' is constructed such that $M \models R(\vec{c})$ if and only if $M' \models R(\vec{c})$. Since the node w has $M \models R(\vec{c})$ while $M' \models \neg R(\vec{c})$ the constant c' must appear at least once in the vector $\vec{c}$ which is what I wanted to explain.

Since we choose c' such that it does not appear in any atomic formula associated to 'free choice' node along $\mathbf{b}$, the question '$R(\vec{c})$' cannot have been left open for the prover. But then all models $M \in \tilde{U}$ which satisfies the conjunctions in A, must satisfy $R(\vec{c})$. Thus M as well as M' satisfy $R(\vec{c})$ i.e. $M \models R(\vec{c})$ and $M' \models R(\vec{c})$ which is a contradiction. Thus we have proved that M' satisfies the conjuction A. This completes the proof of claim C and claim B.

We also have argued that all models $M' \in \tilde{U}$ which satisfies the conjunction A must have $f(\vec{s}) \neq c'$. This contradics the fact that $M' \models f(\vec{s}) = c'$ and shows that our assumption that the prover could win against the adversary's strategy fails. This proves claim A and thus completes the proof of the lemma. □

Combining lemma 4.1 and lemma 4.2 we get:

Lemma 4.3 *Assume there is a model M of T which contain more than n elements. Then each tree resolution refutation $\mathcal{T}$ of $\mathcal{S}_{T,n}$ contains, as a subtree $\mathcal{T}'$, the binary subtree which has hight $(n - k_T^{\text{fun}})/\max(k_T^{\text{rel}}, 1 + k_T^{\text{fun}})$ and size at least $2^{n/\max(k_T^{\text{rel}}, 1+k_T^{\text{fun}})}$. Consequently each tree resolution refutation of $\mathcal{S}_{T,n}$ must have size $\geq 2^{n/\max(k_T^{\text{rel}}, 1+k_T^{\text{fun}})}$.*

Corollary 4.4 *In theorem 3.2 (1) holds if T has an infinite model or if T has a model of size $n' > n$.*

5 The polynomial size upper bound

An example:

Consider the inconsistent theory T which consists of the two axioms:

(1) $\forall x \exists y \forall z \; R(x, y, z)$

(2) $\exists x \forall y \exists z \; \neg R(x, y, z)$

To illustrate the idea behind the general upper bound, we show that asymptotically (i.e when n tends to infinity) $\mathcal{S}_{T,n}$ has polynomial size (in n) tree-resolution refutations.

The language L of the theory T contains a single ternary relation symbol R as its only primitive non-logic symbol. We want to translate the fact that T has no models of size n into an unsatisfiability problem in propositional logic. First we skolemize T and get **(1a)** $R(x, f(x), z)$ and **(2a)** $\neg R(c, y, g(y))$.

Then we rewrite these two formulas as disjunctions of basic atomic formulas and negations of basic atomic formulas: **(1b)** $R(x, y, z) \vee \neg f(x) = y$ and **(2b)** $\neg R(x, y, z) \vee \neg c = x \vee \neg g(y) = z$.

Next add new constants $c_1, c_2, \ldots c_n$ to L. Substitute x, y and z by each combination of the n^3 combinations of c_i, c_j and c_k. Finally inclose the resulting basic atomic formulas in corner quotes:

(1c) $\ulcorner R(c_i, c_j, c_k) \urcorner \;\vee\; \neg \ulcorner f(c_i) = c_j \urcorner$

(2c) $\neg \ulcorner R(c_i, c_j, c_k) \urcorner \;\vee\; \neg \ulcorner c = c_i \urcorner \;\vee\; \neg \ulcorner g(c_j) = c_k \urcorner$

After adding clauses ensuring functions are well behaved (the clauses in $\Gamma_{\leq n}$) we get (letting $r_{ijk} := \ulcorner R(c_i, c_j, c_k) \urcorner, f_{ij} := \ulcorner f(c_i) = c_j \urcorner, g_{ij} := \ulcorner g(c_i) = c_j \urcorner$ and $d_i := \ulcorner c = c_i \urcorner$) the system $\mathcal{S}_{T,n}$:

(1c)$'$ $\{r_{ijk}, \neg f_{ij}\}, \quad i, j, k \in \{1, 2, \ldots, n\}$

(2c)$'$ $\{\neg r_{ijk}, \neg d_i, \neg g_{jk}\}, \quad i, j, k \in \{1, 2, \ldots, n\}$

(i) $\{f_{i1}, f_{i2}, \ldots, f_{in}\}, \quad i \in \{1, 2, \ldots, n\}$

(ii) $\{g_{i1}, g_{i2}, \ldots, g_{in}\}, \quad i \in \{1, 2, \ldots, n\}$

(iii) $\{\neg f_{ij}, \neg f_{ik}\}, \quad j \neq k, i, j, k \in \{1, 2, \ldots, n\}$

(iv) $\{\neg g_{ij}, \neg g_{ik}\}, \quad j \neq k, i, j, k \in \{1, 2, \ldots, n\}$

We know that $\mathcal{S}_{T,n}$ is unsatisfiable (lemma 3.1 + the fact that T has no model of size n). Hence $\mathcal{S}_{T,n}$ has a tree-resolution refutation. We want to show that there exists a polynomial p such that $\mathcal{S}_{T,n}$ for each n has a tree refutation $\mathcal{T}_n$ containing less than $\leq p(n)$ symbols.

Consider **1a** and **2a** again. The axioms are inconsistent so according to Herbrands Theorem (see for example [24] or [10] for more details) there is a unification (namely $x \rightarrow c, y \rightarrow f(c), z \rightarrow g(f(c))$) with makes the clauses $\{R(x, f(x), z)\}$ and $\{\neg R(c, y, g(y))\}$ contradictory in the sense of propositional logic. The substitutions given by the unification leads to the refutation $\mathcal{R}$:

$$\frac{\{\neg R(c, f(c), g(f(c)))\} \quad \{R(c, f(c), g(f(c)))\}}{\emptyset}$$

This refutation serves as a starting point for constructing a refutation $\mathcal{T}_n$ of $\mathcal{S}_{T,n}$. Replace $\mathcal{R}$ with $\mathcal{R}'$:

$$\frac{\{\neg R(x, y, x)\} \cup S \quad \{R(x, y, z)\} \cup S}{S}$$

where $S = \{\neg c = x, \neg f(x) = y, \neg g(y) = z\}$. For each interpretation of c (given by $d_i := \lceil c = c_i \rceil$) we select an interpretation of $f(c)$ (given by $f_{ij} := \lceil f(c_i) = c_j \rceil$) and for each of these interpretations we select an interpretation of $g(f(c))$ (given by $g_{jk} := \lceil g(c_j) = c_k \rceil$). We now build a decision tree $\mathcal{T}$ which refute $\mathcal{S}_{T,n}$ as follows. First, we build a decision tree $\mathcal{T}'$ where each branch either: (i) define an interpretation of $c, f(c)$, and $g(f(c))$ i.e. satisfy $d_i \wedge f_{ij} \wedge g_{jk}$ or (ii) Contain decisions which falsify a clause in **(i)** or **(ii)**. Second, we extend the decision tree $\mathcal{T}'$ as follows: In each leaf where the terms c, $f(c)$ and $g(f(c))$ are interpreted we extend the decision tree by adding a (constant size) decision tree. The decision tree is got by substituting $x = c_i, y = c_j$ and $z = c_k$ into the resolution refutation $\mathcal{R}'$ and turning it into a decision tree. In other words in each leaf where $d_i \wedge f_{ij} \wedge g_{jk}$ holds we ask the question 'r_{ijk}?'. This leads to the following modification of $\mathcal{R}$:

$$\frac{d_i \wedge f_{ij} \wedge g_{jk}}{r_{ijk} \wedge d_i \wedge f_{ij} \wedge g_{jk} \qquad \neg r_{ijk} \wedge d_i \wedge f_{ij} \wedge g_{jk}}$$

Each decision (r_{ijk} or $\neg r_{ijk}$) leads to a violation of a clause in $\mathrm{SAT}_{T,n}$.

As already remarked we can turn the decition tree into a tree-resolution refutation which refuses $\mathrm{SAT}_{T,n}$. The tree has $2n^3 + 2n + 1$ leaves and thus contain $4n^3 + 4n + 1$ clauses. Notice that the clauses in **(iii)** and **(iv)** (which insures that f and g are uni-valued) not are used by the prover and thus does not appear in the resolution refutation.

General upper bound:

Now consider any given inconsistent theory T. We want to show that there exists a polynomial $p(n)$ such that we for each n can refute $\mathcal{S}_{T,n}$ by a tree-resolution refutation $\mathcal{T}_n$ of size $\leq p(n)$.

In general the axioms of T need not be on prenex normal form. Thus there might be different ways of skolemizing the theory. Fix a skolemization and consider the corresponding universal theory T' as well as the corresponding unsatisfiability problem $\mathcal{S}_{T,n}$. Without loss of generality we can assume that the language of T' contains at least one constant c (otherwise we let one of the new constants $c_1, c_2, \ldots, c_n$ play the role of c).

We present the construction in terms of a dialog between an adversary and a prover (in the style of section 4). The adversary claims that $\mathcal{S}_{T,n}$ has a satisfying assignment (which is induced by a model of size n). More specifically the adversary claims that there is a model M with the underlying set $\{c_1, c_2, \ldots, c_n\}$ which induce a satisfying assignment for $\mathcal{S}_{T,n}$. The truth assignment appears by setting each propositional variable of the form $\lceil R(c_{i_1}, c_{i_2}, \ldots, c_{i_k})\rceil$ to true if and only if $R(c_{i_1}, c_{i_2}, \ldots, c_{i_k})$ holds in M and by setting each propositional variable of the form $\lceil f(c_{i_1}, c_{i_2}, \ldots, c_{i_k}) = c_{i_{k+1}}\rceil$ to true if and only if $f(c_{i_1}, c_{i_2}, \ldots, c_{i_k}) = c_{i_{k+1}}$ holds in M. We want to construct a polynomial size winning strategy for the prover.

We assumed that T is inconsistent. Let T' denote the skolemized version of T (used when obtaining $\mathcal{S}_{T,n}$) where all axioms are disjunctions. According the the compactness theorem there is a finite subset $\{\psi_1(\vec{x}), \psi_2(\vec{x}) \ldots \psi_k(\vec{x})\}$ of the set of axioms for T', such that the sentence $\Psi(\vec{x}) := \psi_1(\vec{x}) \wedge \psi_2(\vec{x}) \wedge \ldots \wedge \psi_k(\vec{x})$ is inconsistent (each ψ_j is a quantifier free disjunction of (negations of) basic atomic sentences and each ψ_j has its free variable appear among the variables $\vec{x} = (x_1, x_2, \ldots, x_k)$). According to the completeness theorem the sentence $\exists \vec{x}\ \neg\Psi(\vec{x})$ is provable in logic. Thus using a standard result from proof theory there exist terms t_{ij} such that $\neg\Psi(\vec{t_1}) \vee \neg\Psi(\vec{t_2}) \vee \ldots \vee \neg\Psi(\vec{t_v})$ (where $\vec{t_i} := (t_{i1}, t_{i2}, \ldots, t_{ik})$) is logically valid. This result (which together with an explecit construction is essentially the weak version of Herbrand's theorem discussed in [10]) is an easy consequence of Gentzen's Haupsatz, but it can also be derived purely model-theoretically (see for example [23] or [20]).

Let U denote the (finite) collection of all closed terms and sub-terms of the terms t_{ij}. In the example $\Psi(x, y, z) := (R(x, y, z) \vee \neg f(x) = y) \wedge (\neg R(x, y, z) \vee \neg c = x \vee \neg g(y) = z)$ was the inconsistent sentence. In general when a sentence of the form $\exists x, y, z\ \neg\Psi(x, y, z)$ is logically valid, there exist a number k (some times refered to as the Herbrand Complexity) and closed terms $t_{j1}, t_{j2}, t_{j3}, \quad j = 1, 2, \ldots, k$ such that $\vee_{j=1}^{k} \neg\Psi(t_{j1}, t_{j2}, t_{j3})$ is logically valid. In the example we can choose $k = 1$ since $\neg\Psi(c, f(c), g(f(c)))$ (i.e. $(R(c, f(c), g(f(c))) \vee \neg f(c) = f(c)) \wedge (\neg R(c, f(c), g(f(c))) \vee \neg c = c \vee \neg g(f(c)) = g(f(c))))$ is logically valid. Notice that $U = \{c, f(c), g(f(c))\}$ in the example.

The first part of the provers strategy is to force the adversary to give an interpretation of each closed term (and to each closed sub-term) which appears in U. The terms in U might be nested while the prover can only ask basic atomic questions. Thus if the prover want to make progress she must proceed in a systematic fashion. We define the rank of a closed term inductively: Constants have rank 1. In general if $t_1, t_2, \ldots, t_k$ each has rank $\leq v$ and if f is a k-ary function symbol, then the term $f(t_1, t_2, \ldots, t_k)$ has rank $v + 1$. The first step in the provers strategy is to force the adversary to give interpretations of all constants in U (please do not confuse these constants with the c_j's). If the adversary refuse to give a given constant an interpretation he will eventually violate a clause of the form $\{\lceil d = c_1 \rceil, \lceil d = c_2 \rceil, \ldots, \lceil d = c_n \rceil\}$. Let $d_1, d_2, \ldots, d_v$ denote all constants in U. The prover forces the adversary to interpret these as elements in $\{c_1, c_2, \ldots, c_n\}$. This is done by asking each question $\lceil d_i = c_j \rceil$ where $i = 1, 2, \ldots, v$ and $j = 1, 2, \ldots, n$. Thus after having asked at most $v \times n$ questions the prover have obtained values for each constant in U. Now assume that the prover already have forced the adversary to give an interpretation of each terms in U which has rank $\leq v$. If this includes all (finitely many) terms in U we are done. If there are terms (in U) which are still not interpreted, there must be some terms of rank $v + 1$. The prover force the adversary to select interpretations for each of these terms. The prover can not enquire directly about such a term. Let $f(t_1, t_2 \ldots, t_k)$ denote an arbitrary term of rank $v + 1$. The prover want to force the adversary to give the term a value (in $\{c_1, c_2, \ldots, c_n\}$). The adversary has already given us interpretations of $t_1, t_2 \ldots, t_k$. Say these are $s_1, s_2, \ldots, s_k \in \{c_1, c_2, \ldots, c_n\}$. Then the prover repeat asking questions $\lceil f(s_1, s_2, \ldots, s_k) = c_j \rceil, j = 1, 2, \ldots, n$ until the adversary eventually accepts a value (if he continue to refuse to give a value, he will eventually violate the clause $\{f(\vec{s}) = c_1, f(\vec{s}) = c_2, \ldots, f(\vec{s}) = c_n\} \in \mathcal{S}_{T,n})$. The prover repeat this procedure until each term ($\in U$) of rank $v + 1$ has been given an interpretation. Eventually, the adversary either makes an elementary contradiction, or is forced to give an interpretation of each of the finitely many terms in U.

The adversary is now in great trouble!! He is about to be caught in his net of lies! This will happen within a constant number (i.e. a number independent of n) carefully selected questions.

The prover has forced the adversary to assign values to each term in U. Each (possibly nested) term t_{ij} has been assigned a value $s_{ij} \in \{c_1, c_2, \ldots, c_n\}$. Since the sentence $\neg\Psi(\vec{t_1}) \vee \neg\Psi(\vec{t_2}) \vee \ldots \vee \neg\Psi(\vec{t_v})$ is logically valid, the sentence $\Theta := \neg\Psi(\vec{s_1}) \vee \neg\Psi(\vec{s_2}) \vee \ldots \vee \neg\Psi(\vec{s_v})$ (where $\vec{s_i} := (s_{i1}, s_{i2}, \ldots, s_{ik})$) is also logically valid (still in the sense of predicate logic).

Notice that Θ is build up from basic atomic sentences. We claim Θ is logically valid if each atomic sentence in Θ is inclosed in corner quotes. Or in other words we claim that Θ is logically valid when viewed as a formula in propositional logic. This claim is essentially the weak version of Herbrands Theorem discussed in [10]. The claim also follows directly: Assume there is a truth assignment which makes Θ false. Each basic atomic sentence in Θ has assigned the value 0 or 1. Now (using the construction we used in the proof of lemma 3.1) we can construct a model M with universe $\{1, 2, \ldots, n\}$ such that Θ fails in M. This is a contradiction since Θ is logically valid in the sense for predicate logic.

Since the resolution proof system is complete, there must be a tree resolution refutation $\mathcal{R}$ of $\Psi(\vec{s_1}) \wedge \Psi(\vec{s_2}) \wedge \ldots \wedge \Psi(\vec{s_v})$ with atomic sentences in corner quotes (in the example the refutation $\mathcal{R}$ was given by $\frac{\{r_{ijk}\} \;\; \{\neg r_{ijk}\}}{\emptyset}$). Equivalently there is a decision tree $\mathcal{T}'$ which querring propositional variables leads to a conjunctions of literals which violates a clause i.e. violates $\Psi(\vec{s_j})$ for some $j = 1, 2, \ldots, v$. But the clauses $\Psi(\vec{s_j})$ all appears in $\mathcal{S}_{T,n} (= \mathcal{S}_{T',n})$ so the decision $\mathcal{T}'$ produce a violation of a clause in $\mathcal{S}_{T,n}$. Notice that the size of $\mathcal{T}'$ is independent of n.

Clearly we can build a decision tree from the provers strategy. To show that this decision tree can be bounded by a fixed polynomial, it suffices (this was also noticed in [28]) to show that the number of different possible games (when the adversary varies his strategies) can be bound by a fixed polynomial. Let u denote the number of different terms in U. The adversary have n possible choices of assignment for each term $t \in U$. Besides that the adversary has the freedom to select a path through $\mathcal{R}'$. Thus the number of possible games is bound by λn^u for some $\lambda > 0$. Thus we have shown:

Lemma 5.1 *Let T be an inconsistent theory. Consider $\mathcal{S}_{T,n}$ (with respect to some fixed skolemization T'). If T' has a refutation involving a set U of terms and sub-terms, then $\mathcal{S}_{T,n}$ has a tree-refutation of size $O(n^{|U|})$.*

Now consider a theory T which has no infinite models. According to the compactness theorem, T cannot have arbitrary large finite models (for details see any standard text book in model theory e.g. [20]). Assume T has no models of size $\geq n_0$. Then $T \cup \Gamma_{\geq n}$ is inconsistent for each $n \geq n_0$. By applying lemma 5.1 to $T \cup \Gamma_{\geq n}$ we get:

Corollary 5.2 *In theorem 3.2, (2) holds if T has no infinite model.*

Combining Lemma 4.3, Corollary 4.4, Lemma 5.1 and Corollary 5.2 we get Theorem 3.2.

Proof of Theorem 3.5: Let θ denote a first order sentence which among its relational symbols contain a unary relation symbol U. Assume that the set $\{a \in M : M \models U(a)\}$ is infinite for each model M of θ. One such choice of θ could be the conjunction of $U(1)$, $U(x) \to U(Sx)$, $\neg(S(x) = 1)$ and $Sx = Sy \to x = y$.

Let η be an arbitary universal relational sentence which holds in all models of size $\leq n_0$, but fails in all models of size $\geq n_0 + 1$. Assume that η has no relation symbols in common with θ.

Consider the universal first order sentences η_U where the universal quantifiers are bounded by U (i.e. each quantifier $\forall x$ appears in the context $\forall x(U(x) \to \ldots)$.

We assumed η is satisfiable in some model of size $\leq n_0$, but is unsatisfied in all models of size n_0+1. Clearly $\psi := \theta \wedge \eta_U$ is unsatisfiable for any n. We claim the sequence $\mathcal{S}_{\psi,n}$ require tree-resolution refutations of size $\Omega(n^{n_0})$. To see this let the adversary choose a subset $\hat{U} \subset \{1, 2, \ldots, n\}$ of size n_0, and let the adversary choose a satisfying assignment such that η_U is satisfied (when U has size $\leq n_0$). The adversary makes sure that the elements satisfying U are given interpretations as elements in $\hat{U}$. When adversery is forced to resign (i.e. loose), prover must have forced adversary to claim that at least $n_0 + 1$ concrete elements are in U. Thus at some node v_b (along the branch b) exactly n_0 elements in U are given interpretations as elements in $\hat{U}$. In other words each element in $\hat{U}$ (and no other element) has be given an interpretation as an element b for which $U(b)$ holds. Thus two different sets $\hat{U}$ (each containing n_0 elements) produce different (incompatible) branches. Consequently the provers decision tree must contain at least $\binom{n}{n_0} = \Omega(n^{n_0})$ leafs. Thus $\mathcal{S}_{\psi,n}$ require tree refutations of size $\Omega(n^{n_0})$.

Since there is no total recursive function which given the Gödel number 'η' of a universal relational sentence η, outputs $n_0 := f('\eta')$ such that η has no model of size $\geq n_0$, there is no total recursive function which given input 'ψ', produce an upper bound on the degree n_0 needed to refuse $\mathcal{S}_{\psi,n}$. □

6 Related Results

Assume that ψ is a conjunction of Π_2 sentences, i.e. each sentence in the conjunction is of the form $\forall\vec{x}\exists\vec{y}\ \Theta(x,y)$ where Θ is quantifier-free. If Θ can be written as a single clause we say $\forall\vec{x}\exists\vec{y}\ \Theta(x,y)$ is a *special* Π_2 sentence. It turns out that any conjunction of special Π_2 sentences can be translated into a sequence of (polynomially bounded) satisfiability problems.

Example: Let ψ be the conjunction of
$\forall x,y,z\ \neg(x \prec y) \vee \neg(y \prec z) \vee (x \prec z)$ and $\forall x\ \neg(x \prec x)$ and $\forall x \exists y\ (x \prec y)$. The sentence ψ is saying that $\prec$ defined a partial ordering with no minimal elements.

The sentence is on a special form which allows us to translate it into propositional logic without introducing Skolem functions. More specifically we can write the sentence as the clauses $\Psi_{\text{prop},n}$ consisting of $\{\bar{r}_{ij}, \bar{r}_{jk}, r_{ik}\}$ where $i,j,k \in \{1,2,\ldots,n\}$ are distinct, $\{\bar{r}_{ii}\}$ where $i \in \{1,2,\ldots,n\}$ and $\{r_{i1}, r_{i2}, \ldots, r_{in}\}$ where $i \in \{1,2,\ldots,n\}$.

Had we used our standard translation using Skolem functions we would have got the same clauses except that the clauses $\{r_{i1}, r_{i2}, \ldots, r_{in}\}$ would have been 'skolemised' and replaced by the clauses $\{\bar{f}_{ji}, r_{ij}\}$, $\{f_{i1}, f_{i2}, \ldots, f_{in}\}$ and $\{\bar{f}_{ij}, \bar{f}_{ik}\}$ where $j \neq k$.

The essential difference between the two translations is that our previous translation produce the clauses $\{\bar{f}_{ij}, \bar{f}_{ik}\}$ where $j \neq k$, while these clauses are absent in the 'special' translation. It is clear that the 'special' translation is at least as difficult to refute as the usual translation. Hence, since there are infinite models of ψ, we know that $\mathcal{S}_{\psi,n}$ require tree resolution refutations of size $2^{n/2}$ and thus the same lower bound applies to $\Psi_{\text{prop},n}$. ♠

The above translation works for any conjunction Γ of special Π_2-sentences. The resulting clauses are equivalent to the system $\mathcal{S}_{\Gamma,n}$ if we remove the clauses ensuring that the Skolem functions defines unique values (the clauses (iv),(v) and (vi) in the example of DLO).

Notice however that these axioms not are used in our proof of our polynomial upper bound. To see this notice that if the adversary is allowed to interpret each closed term in more than one way, the provers strategy will still work. This give a heuristic explanation why the polynomial upper in Theorem 3.2 remain valid.

An *extended translation* of a first order sentence ψ in predicate logic, consists of a system $\mathcal{S}_{\psi,n}$ of clauses, which appears by first skolemising ψ (so it becomes a conjunction of universal sentences and possibly also some special Π_2-sentences), and then translating the resulting conjunction of sentences into propositional logic (in the obvious fashion already described). Since a first order sentence has many skolemisations, usually a first order sentence ψ has many extended translations into predicate logic. The upper bound can easily be modified (there is essentially nothing to do, since its the same application of Herbrand Theorem we need) so we get:

Theorem 6.1 *Let ψ be a first order sentence and let $\mathcal{S}_{\psi,n}$ denote any extended translation of ψ into clausal form. Then $\mathcal{S}_{\psi,n}$ is satisfiable if and only if ψ has a model of size n. Further more either $\mathcal{S}_{\psi,n}$ have polynomial size tree resolution refutations (for all sufficiently large n) or there exists a constant $\lambda > 0$ such that $\mathcal{S}_{\psi,n}$ have no tree resolution refutation of size $\leq 2^{\lambda n}$ for any n.*
The second case holds if and only if ψ has an infinite model.

This version of the main theorem, shows that the complexity gap is robust with respect to chosen method of translation.

Finally it should be pointed out that there is one kind of translations which is not covered by our main theorem. This translation arise if one consider sentences ψ in a first order language $L = L(\prec, \ldots)$ which contains a relation symbol $\prec$ which defines a total ordering. In our approach we treat $\prec$ as any other relation symbol. It is however possible to treat $\prec$ as a build-in relation symbol, and let $\lceil c_i \prec c_j \rceil = 1$ if $i < j$ and let $\lceil c_i \prec c_j \rceil = 0$ otherwise. The complexity gap theorem does not cover this kind of translation. This explain why the complexity theorem does not apply directly to for example the mutilated chess board problem which was studied in [15].

7 Further perspectives

Up to this point uniform sequences $\mathcal{S}_{T,n}$ of unsatisfiability problems have been considered. Notice however that our polynomial upper bound was achieved by highly *uniform families of tree-refutations*. This raises a crucial question. Given a propositional proof system P. When does P have the property that uniformly generated (i.e. S_n-generated) sequences of tautologies which have short P-proofs also have short uniformly (used here in an informal sense) generated P-proofs? In this paper we have seen that tree-resolution has this property.

Let T_{total} denote the mathematical theory which is axiomatised by the class of existential sentences ψ that are valid in each finite model (the sentences are written in an infinite language with an arbitrary number of function and relation symbols of each arity). This theory consists of the class of first order formulas which hold in all finite models. Notice that T_{total} is a well-defined theory because the property of being valid in all finite models is closed under logical deduction. The list of sentences in T_{total} forms a complete co-recursively enumerable set. Thus T_{total} is not recursively axiomatisable.

Given a propositional proof system P we let $T_P \subseteq T_{\text{total}}$ denote the collection of existential sentences $\psi \in T_{\text{total}}$ for which the sequence $\mathcal{S}_{\neg\psi,n}$ has polynomial size (or for example $n^{\log(\mathrm{n})^{o(1)}}$-size) P-refutations.

To show a super-polynomial lower bound for the propositional system P it suffices to show that $T_P \neq T_{\text{total}}$. In this paper we showed that when P is tree-resolution then T_P is the minimal theory i.e. just predicate logic. For tree-resolution T_P is recursively axiomatisable (axiomatised by the empty set (!) of axioms over predicate logic). In general for stronger propositional proof systems, T_P need not be recursively axiomatisable. It is a mistake to equate T_P with $I\Delta_0(\alpha)$ (resp. V_1^1 or U_1^1) when P is polynomial size bounded depth Frege proofs (resp. polynomial size extended Frege proofs or $n^{\log(n)^{o(1)}}$-size Frege proofs). A detailed description of these systems can be found in [21]. The theory T_P is axiomatised by purely existential axioms and thus it behaves differently from the theories $I\Delta_0(\alpha)$, V_1^1 or U_1^1 which are all recursively axiomatisable by universal axioms.

It is possible to formulate various interesting conjectures and questions concerning general properties of the axiomatisations of T_P. Is T_P is recursively axiomatisable when P is dag-resolution or when P is the Frege system? I conjecture that T_P is not recursively axiomatisable when P is the extended Frege system. In general it seems safe to conjecture that:

Conjecture 7.1 *For any propositional proof system* $T_P \neq T_{\text{total}}$.

Using Cook's and Reckhow's reformulation of the NP $\neq$ co-NP question [14] we get:

Proposition 7.2 *The conjecture implies NP $\neq$ co-NP.*

The main aim in this section is to define the theory T_P (where P is an arbitrary propositional proof system). Theorem 3.2 can now be stated as follows:

Theorem 7.3 *Let P denote tree-resolution and let $T_P(f) \subseteq T_{\text{total}}$ denote the theory of principles $\psi \in T_{\text{total}}$ which have $f(n)$-size tree proofs (refutations). The theory $T_P(f)$ is well-defined as well as well-behaved. Furthermore $T_P(f)$ is the same theory (namely predicate logic) for any super-polynomial but sub-exponential function $f(n)$.*

8 General considerations

Consider the theory $T_P(f)$ defined in the previous section. When the growth rate of f changes, the theory $T_P(f)$ might, of course, also change. The question is whether this change happens in a continuous manner or - as we have shown for tree-resolution - happens in a jump.

Let us briefly compare our translation method with another important theoretical method for generating satisfiability problems. This method (which is outside the scope of this paper) is to consider randomly chosen 3-satisfiability problems and to consider the case where the ratio c of clauses and variables is kept constant, while the number of variables tends to infinity. Experiments suggest that there is a phase transition near $c = c_{phase} \approx 4.258\ldots$ [13]. Experimentally it is found that virtually all problems with $c > c_{phase}$ are unsatisfiable, while virtually all problems with $c < c_{phase}$ are satisfiable. Given a deductive powerful propositional system P (like the Extended Frege system) it seems reasonable to expect that there exists some constant c_{poly} such that with probability converging to 1 any randomly chosen 3-sat problem with ratio $c'' \geq c_{\text{poly}}$ have polynomial size refutations. Assume also that a randomly chosen problem with $c = c_{prase}$ with probability converging to 1 requires P-refutations of a size bounded from below by some exponential expression. If Hardy's remark (which we discussed in the introduction) applies to this situation, for each c (with $c_{phase} \leq c \leq c_{\text{poly}}$) there should be a logarithmic-exponential expression estimating the optimal size of P refutations. Any such expression, (by which $P(n)\exp(\lambda n/(c - c_{\text{poly}}))$ for $c_{phase} \leq c \leq c_{\text{poly}}$ is just one of infinitely many possibilities) must have a complexity gap at some c'. So (assuming that Hardy's remark applies) there exists at least one phase transition i.e. constant c_P such that: For $c_{phase} \leq c < c_P$ a randomly chosen 3SAT problem almost certainly only has long P-refutations (of size bounded from below by a concrete exponential expression). For $c_P < c$ there almost certainly exists a short (bound from above with sub-exponential logarithmic-exponential expression) P-refutation. In such a case, where the threshold c_P is sharp, it seems fair to say that a complexity gap occurs. Of course the situation could be very complicated with various phase transitions and thresholds corresponding to different complexity classes etc.

These considerations touches a fundamental question. As already suggested it seems to be an empirical fact that only a relatively small number of complexities appears in practice (like $\theta(1)$, $\theta(\sqrt{\log(n)/\text{loglog}(n)})$, $\theta(\log(n)/\text{loglog}(n))$, $\theta(n)$, $\theta(nlogn)$, $\theta(n^2)$, $2^{n^{\theta(1)}}$)). These complexities can be all given by logarithmic-exponential expressions. Theoretically, virtually any complexity is possible, so why do so the same complexities arise again and again? This is striking because the number of different problems is significantly more extensive than the above small finite list. One feature of real world computational problems is that they, in some sense, involve computational problems which are the *same* except for size. It would, for example, be highly unnatural to consider a computational problem where certain lists have to be sorted for some values of n while certain bin-packing problems have to be solved for other values of n. Uniformity is clearly a feature of real world problems as we meet them in theoretical computer science.

I hope the reader will forgive these pure speculations, but if Hardy's remark is widely valid then we will expect that any *uniformly* (here used informally) given computational problem will have a worst case complexity which belongs to a list of discrete possibilities (all given by logarithmic-exponential expressions). Perhaps all of these are the shadows of a master theorem. A Theorem which surely must be far beoynd current techniques and which states that a large class of uniform complexity questions can only have certain discrete answers. Or perhaps there is no such theorem and the phenomena only reflect the limitations in our methods.

In the setting of propositional logic perhaps $T_P(f)$ always has a discrete set of of jumps. Clearly (using Cook's and Reckhow's result [14]) if there is a complexity gap for any propositional system P we must have NP $\neq$ co-NP.

For a given propositional system P an interesting project is to give criteria for when a proposition Ψ leads to unsatisfiability problems $\mathcal{S}_{\Psi,n}$ which require super polynomial size P-refutations. Consider for example the NS-proof system (over fields of characteristic 0). This is a very interesting propositional proof system which has been studied intensively in recent years. The system was first introduced in [4] and has many nice features [12]. We finish the paper by showing that the Nullstellensatz proof system proves the following version of the pigeonhole principle.

For fixed $n \in \mathbf{N}$ consider the class Poly$_n$ of polynomials in the variables x_{ij} where $i \in \{0, 1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, n\}$.

Consider the following polynomial equations:

$$(\sum_{j=1, j\neq i}^{n} x_{ij}) + x_{ii} - 1 = 0 \text{ for } i = 1, 2, \ldots, n, \quad (\sum_{j=1}^{n} x_{0j}) - 1 = 0.$$

$$(\sum_{j=1, j\neq i}^{n} x_{ji}) + x_{ii} + x_{0i} - 1 = 0 \text{ for } i = 1, 2, \ldots, n.$$

These equations have no 0/1-solution as such a solution would define a bijection from $\{0, 1, \ldots, n\}$ onto $\{1, 2, \ldots, n\}$. Actually we show:

Proposition 8.1 *The equations do not have any solutions over any ring.*

Proof: Notice that

$$\sum_{i=1}^{n}((\sum_{j=1,j\neq i}^{n} x_{ij})+x_{ii}+x_{0i}-1)-\sum_{i=1}^{n}((\sum_{j=1,j\neq i}^{n} x_{ji})+x_{ii}-1)-(\sum_{j=1}^{n} x_{0j})-1=1$$

and that 1 thus can be written as a linear combination of the polynomials which appear in the polynomial equations. □

The tautologies for the usual pigeonhole principle is an extension of the above equations. Besides the equations above, they include:

$x_{ij}^2 - x_{ij} = 0$, where $i \in \{0,1,2,\dots,n\}$ and $j \in \{1,2,\dots,n\}$.

$x_{ij}x_{ik} = 0$ for $i \in \{0,1,2,\dots,n\}$ and $j,k \in \{1,2,\dots,n\}$ with $j \neq k$.

$x_{ji}x_{ki} = 0$ for $i \in \{1,2,\dots,n\}$ and $j,k \in \{0,1,2,\dots,n\}$ with $j \neq k$.

It is well known that the bijective pigeonhole principle requires exponential size bounded depth Frege proofs [6]. Thus we have:

Theorem 8.2 *There exists a sequence of tautologies which has linear size Nullstellensatz proofs, but requires exponential size bounded depth Frege Proofs.*

This shows that the strength of the NS-proof system is incompatible to Bounded Depth Frege. It also shows that the complexity gap for the NS-proof system does not take place at the same place as for tree-resolution (i.e. $T_{\mathrm{TR}} \subset T_{\mathrm{NS}}$ with $T_{\mathrm{TR}} \neq T_{\mathrm{NS}}$). An exact characterisation of T_{NS}, i.e. an exact characterisation of the class of ψ's for which $\mathcal{S}_{\psi,n}$ requires polynomial size NS-refutations is open. Is the theory T_{NS} recursively axiomatisable? Also the size of the complexity jump for the NS-proof system is open although some special cases has been settled [35]. The same questions are open for the bounded depth Frege proof system. Does the cutting plane propositional proof system have a complexity gap? Does unrestricted resolution? Does the unrestricted resolution have a complexity gap from polynomial to exponential? What is the characterisation of first order sentences ψ for which $\mathcal{S}_{\psi,n}$ has polynomial size resolution proofs (refutations)?

Acknowledgements I would like to acknowledge Alasdair Urquhart for suggesting rewriting [33]. The main result in [33] is identical to the main result in the revised version with exactly the same constants. However the presentation has been improved and although the idea behind all proofs is the same as originally intended, most proofs have been given a substantially different presentation. I would also like to thank Alasdair Urquhart for generously allowing me to use and include material from [41]. And I would like to thank Johan Håstad, as well as a number of anonymous referees, for their many helpful comments. These comments also helped improving the presentation. Finally I would like to thank Peter Bro Miltersen and Ulrich Kohlenbach for some useful comments concerning an earlier version of the paper.

References

[1] Ajtai, M.: The complexity of the pigeonhole principle. In 29th Annual Symposium on Foundations of Computer Science, IEEE (1988) pp. 346-355.

[2] Ajtai, M.: The complexity of the pigeonhole principle. Combinatorica, 14(4) (1994) 417-433.

[3] Ajtai, M.: The independence of the modulo p counting principles. In Proceedings of the 26th ACM STOC, (1994) 402-411.

[4] Beame, P., Impagliazzo, R., Krajicek, J., Pitassi, T., Pudlak, P.: Lower bounds on Hilbert's Nullstellensatz and propositional proofs. Proceedings of the London Mathematical Society **73(3)** (1996) 1-26.

[5] Beame, P.,Karp R., Pitassi T., Saks M.: On the complexity of Unsatisfiability Proofs for Random k-CNF Formulas. STOC 98. The 30th ACM STOC, (1998)

[6] Beame, P.,Impagliazzo, R., Krajicek, J., Pitassi,T.,Pudlak,P., Woods,A.: Exponential lower bounds for the pigeonhole principle, In the proceedings of the 24th ACM STOC (1992) 200-221

[7] Beame, P., Riis, S.: More on the relative strength of counting principles. In: Proceedings of the DIMACS workshop on Feasible Arithmetic and Complexity of Proofs, (1996)

[8] Buss, S.: The propositional pigeonhole principle has polynomial size Frege proofs, J. of Symbolic Logic, 52 (1987) 916-927.

[9] Buss, S.: Propositional consistency proofs, Annals of Pure and Applied Logic 52 (1991) 3-29.

[10] Buss, S: Introduction to Proof Theory. In: Buss, S., (ed.), Handbook of Proof Theory, pp. 1-78. Elsevier 1998.

[11] Buss, S., Pitassi, T.: Resolution and the weak Pigeonhole Principle (notes).

[12] Buss, S., Krajicek, j,. Pitassi, T., Razborov, A., Sergal, J.: Polynomial bound on Nullstellensatz for counting principles. To appear in Computational Complexity (1997)

[13] Crawford, J., Auton, L.: Experimental Results on the Crossover Point in Random 3SAT; preprint (1996)

[14] Cook, S., Reckhow, R.: The relative efficiency of propositional proof systems, Journal of Symbolic Logic, 44 (1979) 36-50.

[15] Dantchev, S., Riis, S. Planar tautologies are hard for Resolution. http://www.dcs.qmw.ac.uk/ smriis/planar.ps

[16] Galil, Z.: On the complexity of regular resolution and the Davis-Putnam procedure. Theoretical Computer Science, 4 (1977) 23-46.

[17] Kearns, Vazirani: An Introduction to Computational Learning Theory, MIT Press, (1994)

[18] Haken, A.: The intractability of resolution, Theoretical Computer Science, 39 (1985), 297-308.

[19] Hardy, G.H.; Orders of Infinity, Cambridge University Press, 3rd edition (1954)

[20] Hodges, W.: Model Theory, Cambridge University Press, Encyclopedia of Mathematics and its applications, Vol 42 (1993).

[21] Krajicek, J.:Bounded Arithmetic, propositional logic, and complexity theory, Encyclopedia of Mathematics and Its Applications, Vol. 60, Cambridge University Press (1995)

[22] Krajicek, J.: On the degree of ideal membership proofs from uniform families of polynomials over a finite field (manuscript)

[23] Kreisel, G., Krivine, J.-L.: Elements of Mathematical logic: model theory (Revised second ed.). Studies in Logic and the Foundations of Mathematics. North-Holland (1971)

[24] Leitsch, A.: The resolution Calculus. Book in the series of Texts in Theoretical Computer Science, Ed. Brauer, W., Rozenberg, G., Salomaa, A. Springer / Heidelberg (1996)

[25] Lovasz, L. Naor,M., Newman,I.,Wigderson,A. Search problems in the decision tree model. In Proceedings of the 32th ACM STOC, (1991) 576-585

[26] Pudlak, P.: On the length of proofs of finitistic consistency statements in first order theories, in: J.B. Paris et al., eds, Logic Colloquium '84 North-Holland, Amsterdam (1986) 165-196

[27] Pudlak, P.: Improved bounds to the lengths of proofs of finitistic consistency statements, in: Logic and combinatorics, Contemporary Math. 65 (Amer. Math. Soc., Providence, RI (1987) 309-331.

[28] Pudlak, P.: Proofs as games. American Mathematical Monthly, to appear.

[29] Riis, S.: Making infinite structures finite in models of Second Order Bounded Arithmetic. In: Arithmetic, proof theory and computorial complexity, 289-319, Oxford: Oxford University Press 1993

[30] Riis, S.: Independence in Bounded Arithmetic. DPhil dissertation, Oxford University (1993)

[31] Riis, S.: Count(q) does not imply Count(p) Annals of Pure and Applied Logic, 90(1-3) (1997) 1-56

[32] Riis, S.: Count(q) versus the pigeonhole principle. Archive for Mathematical Logic **36** (1997) 157-188

[33] Riis, S.: A complexity gap for tree-resolution. BRICS RS-99-29.

[34] Riis, S., Sitharam, M: Generating hard tautologies using logic and the symmetric group. BRICS RS-98-19.

[35] Riis, S., Sitharam, M: Uniformly Generated Submodules of Permutation Modules. BRICS RS-98-20. Accepted for publication in: Journal of pure and applied Algebra.

[36] Riis, S., Sitharam, M: (manuscript in preparation). Incomplete version appear as: Non-constant Degree Lower Bounds imply Linear Degree Lower Bounds, Technical report TR97-048 of the *Electronic Colloquium*

on Computational Complexity, http://www.eccc.uni-trier.de/pub/eccc (1997)

[37] Simpson,S.: Unprovable theorem and fast growing functions. Contempora Mathematics Vol 65 (1987) 359-394

[38] Tseitin, G.S.: On the complexity of derivations in the propositional Calculus. In A.O. Slisenko, editor, Studies in Constructive Mathematics and Mathematical Logic, Part II (1968)

[39] Urquhart, A.: Hard examples for resolution. Journal of the ACM, 34(1) (1987) 209-219.

[40] Urquhart, A.: The Complexity of Propositional Proofs. The Bulletin of Symbolic Logic, 1 (1995) 425-467

[41] Urquhart, A.: Riis's Complexity Gap for Tree Resolution. Distributed notes.