A Survey of Lower Bounds for Satisfiability and Related Problems

A Survey of Lower Bounds for Satisfiability and Related Problems

Dieter van Melkebeek

University of Wisconsin
Madison
WI 53706
USA
dieter@cs.wisc.edu

now
the essence of knowledge

Boston – Delft

Foundations and Trends® in Theoretical Computer Science

Published, sold and distributed by:
now Publishers Inc.
PO Box 1024
Hanover, MA 02339
USA
Tel. +1-781-985-4510
www.nowpublishers.com
sales@nowpublishers.com

Outside North America:
now Publishers Inc.
PO Box 179
2600 AD Delft
The Netherlands
Tel. +31-6-51115274

The preferred citation for this publication is D. van Melkebeek, A Survey of Lower Bounds for Satisfiability and Related Problems, Foundation and Trends® in Theoretical Computer Science, vol 2, no 3, pp 197–303, 2006

ISBN: 978-1-60198-084-7

Foundations and Trends® in
Theoretical Computer Science
Volume 2 Issue 3, 2006

Editorial Board

Editorial Scope

Foundations and Trends® in Theoretical Computer Science will publish survey and tutorial articles in the following topics:

- Algorithmic game theory
- Computational algebra
- Computational aspects of combinatorics and graph theory
- Computational aspects of communication
- Computational biology
- Computational complexity
- Computational geometry
- Computational learning
- Computational Models and Complexity
- Computational Number Theory
- Cryptography and information security
- Data structures
- Database theory
- Design and analysis of algorithms
- Distributed computing
- Information retrieval
- Operations Research
- Parallel algorithms
- Quantum Computation
- Randomness in Computation

Information for Librarians

Foundations and Trends® in Theoretical Computer Science, 2006, Volume 2, 4 issues. ISSN paper version 1551-305X. ISSN online version 1551-3068. Also available as a combined paper and online subscription.

Foundations and Trends® in
Theoretical Computer Science
Vol. 2, No. 3 (2006) 197–303
© 2007 D. van Melkebeek
DOI: 10.1561/0400000012

now
the essence of knowledge

A Survey of Lower Bounds for Satisfiability and Related Problems

Dieter van Melkebeek[*]

University of Wisconsin, 1210 W. Dayton St., Madison, WI 53706, USA, dieter@cs.wisc.edu

Abstract

Ever since the fundamental work of Cook from 1971, satisfiability has been recognized as a central problem in computational complexity. It is widely believed to be intractable, and yet till recently even a linear-time, logarithmic-space algorithm for satisfiability was not ruled out. In 1997 Fortnow, building on earlier work by Kannan, ruled out such an algorithm. Since then there has been a significant amount of progress giving non-trivial lower bounds on the computational complexity of satisfiability. In this article, we survey the known lower bounds for the time and space complexity of satisfiability and closely related problems on deterministic, randomized, and quantum models with random access. We discuss the state-of-the-art results and present the underlying arguments in a unified framework.

[*] Research partially supported by NSF Career Award CCR-0133693.

Contents

1

Introduction

Satisfiability is the problem of deciding whether a given Boolean formula has at least one satisfying assignment. It is the first problem that was shown to be NP-complete, and is possibly the most commonly studied NP-complete problem, both for its theoretical properties and its applications in practice. Complexity theorists widely believe that satisfiability takes exponential time in the worst case and requires an exponent linear in the number of variables of the formula. On the other hand, we currently do not know how to rule out the existence of even a linear-time algorithm on a random-access machine. Obviously, linear time is needed since we have to look at the entire formula in the worst case. Similarly, we conjecture that the space complexity of satisfiability is linear but we have yet to establish a space lower bound better than the trivial logarithmic one.

Till the late 1990s it was even conceivable that there could be an algorithm that would take linear time and logarithmic space to decide satisfiability! Fortnow [19], building on earlier techniques by Kannan [31], developed an elegant argument to rule out such algorithms. Since then a wide body of work [3, 17, 18, 20, 22, 37, 56, 39, 41, 58, 59, 61] have strengthened and generalized the results to lead to a rich variety of lower bounds when one considers a nontrivial combination of time and space complexity. These results form the topic of this survey.

We now give some details about the evolution of the lower bounds for satisfiability. Fortnow's result is somewhat stronger than what we stated above — it shows that satisfiability cannot have both a linear-time algorithm and a (possibly different) logarithmic-space algorithm. In fact, his argument works for time bounds that are slightly super-linear and space bounds that are close to linear, and even applies to co-nondeterministic algorithms.

Theorem 1.1 (Fortnow [19]). For every positive real ϵ, satisfiability cannot be solved by both

(i) a (co-non)deterministic random-access machine that runs in time $n^{1+o(1)}$ and
(ii) a (co-non)deterministic random-access machine that runs in polynomial time and space $n^{1-\epsilon}$.

In terms of time–space lower bounds, Fortnow's result implies that there is no (co-non)deterministic algorithm solving satisfiability in time $n^{1+o(1)}$ and space $n^{1-\epsilon}$. Lipton and Viglas [20, 37] considered deterministic algorithms with smaller space bounds, namely polylogarithmic ones, and managed to establish the first time–space lower bound where the running time is a polynomial of degree larger than one. Their argument actually works for subpolynomial space bounds, i.e., for space $n^{o(1)}$. It shows that satisfiability does not have a deterministic algorithm that runs in $n^{\sqrt{2}-o(1)}$ steps and uses subpolynomial space.[1] Fortnow and van Melkebeek [20, 22] captured and improved both earlier results in one statement, pushing the exponent in the Lipton–Viglas time–space lower bound from $\sqrt{2} \approx 1.414$ to the golden ratio $\phi \approx 1.618$. Williams [59, 61] further improved the latter exponent to the current record of $2\cos(\pi/7) \approx 1.801$, although his argument no longer captures Fortnow's original result for deterministic machines.

[1] The exact meaning of this statement reads as follows: For every function $f : \mathbb{N} \to \mathbb{N}$ that is $o(1)$, there exists a function $g : \mathbb{N} \to \mathbb{N}$ that is $o(1)$ such that no algorithm for satisfiability can run in time $n^{\sqrt{2}-g(n)}$ and space $n^{f(n)}$.

Theorem 1.2 (Williams [61]). Satisfiability cannot be solved by a deterministic random-access machine that runs in time $n^{2\cos(\pi/7)-o(1)}$ and space $n^{o(1)}$.

The following — somewhat loaded — statement represents the state-of-the-art on lower bounds for satisfiability on deterministic machines with random access.

Theorem 1.3 (Master Theorem for deterministic algorithms). For all reals c and d such that $(c-1)d < 1$ or $cd(d-1) - 2d + 1 < 0$, there exists a positive real e such that satisfiability cannot be solved by both

(i) a co-nondeterministic random-access machine that runs in time n^c and
(ii) a deterministic random-access machine that runs in time n^d and space n^e.

Moreover, the constant e approaches 1 from below when c approaches 1 from above and d is fixed.

Note that a machine of type (ii) with $d < 1$ trivially fails to decide satisfiability as it cannot access the entire input. A machine of type (i) with $c < 1$ can access the entire input but nevertheless cannot decide satisfiability. This follows from a simple diagonalization argument which we will review in Chapter 3 because it forms an ingredient in the proof of Theorem 1.3. Note also that a machine of type (ii) is a special case of a machine of type (i) for $d \leq c$. Thus, the interesting values of c and d in Theorem 1.3 satisfy $d \geq c \geq 1$.

Theorem 1.3 applies when c and d satisfy a disjunction of two conditions. For values of $d > 2$ the condition $(c-1)d < 1$ is less stringent than $cd(d-1) - 2d + 1 < 0$; for $d < 2$ the situation is the other way around. See Figure 1.1 for a plot of the bounds involved in Theorem 1.3. We can use the first condition to rule out larger and larger values of d for values of c that get closer and closer to 1 from above. Thus, Fortnow's

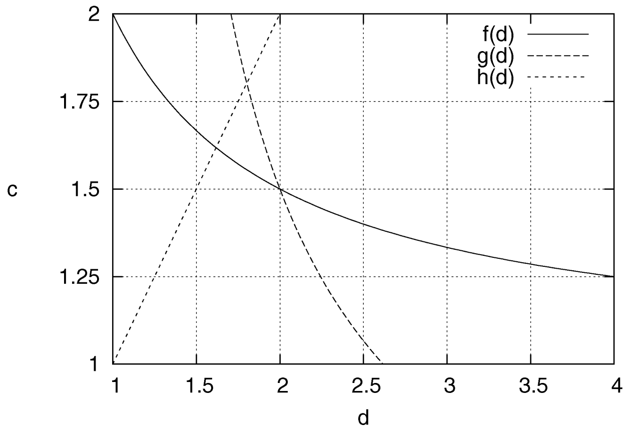

Fig. 1.1 Bounds in the Master Theorem for deterministic algorithms: $f(d)$ solves $(c-1)d = 1$ for c, $g(d)$ solves $cd(d-1) - 2d + 1 = 0$ for c, and $h(d)$ is the identity.

result for deterministic machines is a corollary to Theorem 1.3. The second condition does not hold for large values of d for any $c \geq 1$, but yields better time lower bounds for subpolynomial-space algorithms. We can obtain time–space lower bounds from Theorem 1.3 by setting $c = d$; in that case we can omit the part from the statement involving the machine of type (i) as it is implied by the existence of a machine of type (ii). The first condition thus yields a time lower bound of $n^{d-o(1)}$ for subpolynomial space, where $d > 1$ satisfies $d(d-1) = 1$, i.e., for d equal to the golden ratio $\phi \approx 1.618$. The second condition leads to a time lower bound of $n^{d-o(1)}$ for subpolynomial space, where $d > 1$ satisfies $d^2(d-1) - 2d + 1 = 0$; the solution to the latter equation equals the above mysterious constant of $2\cos(\pi/7) \approx 1.801$, which is larger than ϕ. Thus, the Master Theorem captures Theorem 1.2 as well.

The successive improvements of recent years beg the question how far we can hope to push the time–space lower bounds for satisfiability in the near future. On the end of the spectrum with small space bounds, there is a natural bound of 2 on the exponent d for which the current techniques allow us to prove a time lower bound of n^d for algorithms solving satisfiability in logarithmic space. We will discuss this bound in Section 4.1 and its reachability in Chapter 9. On the end of the spectrum with small time bounds, the quest is for the largest exponent e

such that we can establish a space lower bound of n^e for any algorithm solving satisfiability in linear time. The techniques presented in this survey critically rely on sublinear space bounds so we cannot hope to reach $e = 1$ or more along those lines. Note that sublinear-space algorithms for satisfiability are unable to store an assignment to the Boolean formula.

All the known lower bounds for satisfiability on deterministic random-access machines use strategies similar to one pioneered by Kannan in his early investigations of the relationship between nondeterministic and deterministic linear time [31]. The arguments really give lower bounds for nondeterministic linear time; they translate to lower bounds for satisfiability by virtue of the very efficient quasi-linear reductions of nondeterministic computations to satisfiability. The same type of reductions exist to many other NP-complete problems — in fact, to the best of my knowledge, they exist for all of the standard NP-complete problems. Thus, the lower bounds for satisfiability as stated in Theorem 1.3 actually hold for all these problems. In Section 4.2, we discuss how the underlying arguments can be adapted and applied to other problems that are closely related to satisfiability, such as the cousins of satisfiability in higher levels of the polynomial-time hierarchy and the problem of counting the number of satisfying assignments to a given Boolean formula modulo a fixed number.

Lower bounds for satisfiability on deterministic machines relate to the P-versus-NP problem. Similarly, in the context of the NP-versus-coNP problem, one can establish lower bounds for satisfiability on co-nondeterministic machines, or equivalently, for tautologies on nondeterministic machines. The statement of Theorem 1.3 partially realizes such lower bounds because the machine of type (i) is co-nondeterministic; all that remains is to make the machine of type (ii) co-nondeterministic, as well. In fact, Fortnow proved his result for co-nondeterministic machines of type (ii). Similar to the deterministic case, Fortnow and van Melkebeek [20, 22] improved the time lower bound in this version of Fortnow's result from slightly super-linear to a polynomial of degree larger than 1. In terms of time–space lower bounds on the large-time end of the spectrum, their result yields a time lower bound of $n^{\sqrt{2}-o(1)}$ for subpolynomial space nondeterministic machines that decide tautologies. Diehl et al. [18] improved the exponent in the latter result from $\sqrt{2} \approx 1.414$ to $\sqrt[3]{4} \approx 1.587$ but their proof does not yield nontrivial results at the end of the spectrum with space bounds close to linear.

Theorem 1.4 (Diehl–van Melkebeek–Williams [18]). Tautologies cannot be solved by a nondeterministic random-access machine that runs in time $n^{\sqrt[3]{4}-o(1)}$ and space $n^{o(1)}$.

The following counterpart to Theorem 1.3 captures all the known lower bounds for tautologies on nondeterministic machines with random access.

Theorem 1.5 (Master Theorem for nondeterministic algorithms). For all reals c and d such that $(c^2 - 1)d < c$ or $c^2 d < 4$, there exists a positive real e such that tautologies cannot be solved by both

(i) a nondeterministic random-access machine that runs in time n^c and
(ii) a nondeterministic random-access machine that runs in time n^d and space n^e.

Moreover, the constant e approaches 1 from below when c approaches 1 from above and d is fixed.

Similar to the deterministic setting, the interesting values in Theorem 1.5 satisfy $d \geq c \geq 1$. The hypothesis is the disjunction of two conditions. See Figure 1.2 for a plot of the bounds involved. The first condition is binding for larger values of d and allows us to derive Fortnow's result in full form. The second condition is binding for smaller values of d, which includes the range in which the hypothesis holds for $c = d$. The first condition yields a time lower bound of $n^{d-o(1)}$ for subpolynomial space, where $d > 1$ satisfies $d(d^2 - 1) = d$, i.e., for $d = \sqrt{2}$. The second condition leads to such a lower bound for $d > 1$ satisfying $d^3 = 4$, yielding Theorem 1.4.

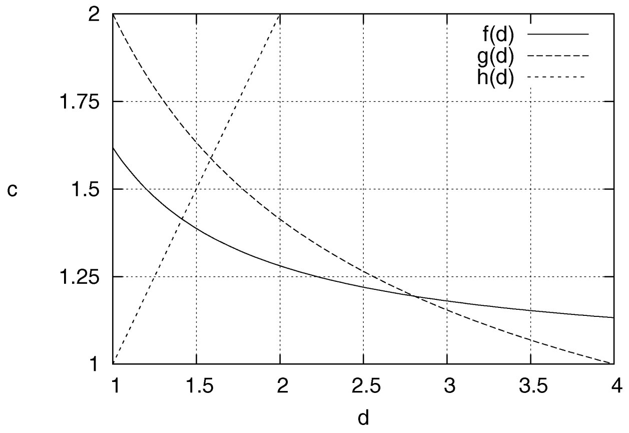

Fig. 1.2 Bounds in the Master Theorem for nondeterministic algorithms: $f(d)$ solves $(c^2 - 1)d = c$ for c, $g(d)$ solves $c^2 d = 4$ for c, and $h(d)$ is the identity.

Theorems 1.3 and 1.5 can be viewed as the first two in a sequence where the machines of type (ii) can have more and more alternations. We will not pursue this sequence any further in full generality but the case of small values for c plays a role in the lower bounds for satisfiability on “somewhat-nonuniform” models, which we discuss next.

Complexity theorists do not think that nonuniformity helps in deciding satisfiability. In particular, we conjecture that satisfiability requires circuits of linear-exponential size. At the same time, we cannot rule out that satisfiability has linear-size circuits.

Time–space lower bounds for deterministic machines straightforwardly translate into size-width lower bounds for sufficiently uniform circuits, and into depth-logarithm-of-the-size lower bounds for sufficiently uniform branching programs. Lower bounds for (co)nondeterministic machines similarly imply lower bounds for very uniform (co)nondeterministic circuits. Logarithmic-time uniformity trivially suffices for all of the above results to carry over without any changes in the parameters. We currently do not know of any interesting lower bounds for fully nonuniform general circuits. However, modulo some deterioration of the parameters, we can relax or even eliminate the uniformity conditions in some parts of Theorems 1.3 and 1.5. This leads to lower bounds with relatively weak uniformity conditions in a few models of interest.

Fortnow showed how to apply his technique to logspace-uniform NC^1-circuits [19]. Allender et al. [3] extended this result to logspace-uniform SAC^1-circuits and their negations. van Melkebeek [39] derived all these circuit results as instantiations of a general theorem, and showed directly that in each case $\mathrm{NTS}(n^{O(1)}, n^{1-\epsilon})$-uniformity for a positive constant ϵ suffices, where $\mathrm{NTS}(t,s)$ refers to nondeterministic computations that run in time t and space s. We can further relax the uniformity condition from nondeterministic to alternating computations of the same type with a constant number of alternations, i.e., to $\Sigma_k\mathrm{TS}(n^{O(1)}, n^{1-\epsilon})$-uniformity for arbitrary constant k. See Section 2.2 for the precise definitions of the complexity classes and uniformity conditions involved.

We discuss “somewhat-nonuniform” versions of Theorems 1.3 and 1.5 in Chapter 6. Here we suffice with the corresponding statement for alternating machines when c ranges over values close to 1, since this setting allows us to capture all the above results.

Theorem 1.6 (Somewhat-nonuniform algorithms). For every nonnegative integer k, every real d, and every positive real ϵ, there exists a real $c > 1$ such that satisfiability cannot both

(i) have $\Sigma_k\mathrm{TS}(n^d, n^{1-\epsilon})$-uniform co-nondeterministic circuits of size n^c and
(ii) be in $\Sigma_k\mathrm{TS}(n^d, n^{1-\epsilon})$.

For certain types of circuits, part (i) implies a uniform algorithm for satisfiability that is efficient enough so that we do not need to state (ii). In particular, we obtain the following corollary to the proof of Theorem 1.6.

Corollary 1.1. For every nonnegative integer k and positive real ϵ, satisfiability cannot be solved by $\Sigma_k\mathrm{TS}(n^{O(1)}, n^{1-\epsilon})$-uniform families of any of the following types: circuits of size $n^{1+o(1)}$ and width $n^{1-\epsilon}$, SAC^1-circuits of size $n^{1+o(1)}$, or negations of such circuits.

Recall that SAC^1-circuits are circuits of logarithmic depth with bounded fan-in ANDs, unbounded fan-in ORs, and negations only on the inputs. NC^1-circuits of size $n^{1+o(1)}$ are a special type of SAC^1-circuits of size $n^{1+o(1)}$. Negations of SAC^1-circuits are equivalent to circuits of logarithmic depth with bounded fan-in ORs, unbounded fan-in ANDs, and negations only on the inputs.

There is another direction in which we can extend the lower bounds to a nonuniform setting. Tourlakis [56] observed that the arguments of Fortnow and of Lipton–Viglas carry through when the machines involved receive subpolynomial advice. The same holds for almost all the results stated in this survey. We refer to Section 3.1 and the end of Chapter 6 for more details.

Other models of computation that capture important capabilities of current or future computing devices include randomized and quantum machines. To date we know of no nontrivial lower bounds for satisfiability on such models with two-sided error but we do have interesting results for problems that are somewhat harder than satisfiability.

In the setting of randomized computations with two-sided error, the simplest problem for which we can prove nontrivial lower bounds is Σ_2SAT, the language consisting of all valid Σ_2-formulas. Σ_2SAT constitutes the equivalent of satisfiability in the second level of the polynomial-time hierarchy.

At first glance, it might seem that results from space-bounded derandomization let us derive time–space lower bounds for randomized algorithms as immediate corollaries to time–space lower bounds for deterministic algorithms. In particular, assuming we have a randomized algorithm that solves satisfiability in logarithmic space and time n^d, Nisan's deterministic simulation [46] yields a deterministic algorithm for satisfiability that runs in polynomial time and polylogarithmic space. However, even for $d = 1$, the degree of the polynomial is far too large for this simulation to yield a contradiction with known time–space lower bounds for deterministic algorithms.

At the technical level, the arguments for satisfiability in the deterministic setting do not carry over to the setting of randomized algorithms with two-sided error. The difficulty is related to the fact that we know efficient simulations of randomized computations with two-sided error in the second level of the polynomial-time hierarchy but not in the first level. Roughly speaking, this is why we have results for Σ_2SAT but not for satisfiability itself. Diehl and van Melkebeek [17] proved the first lower bound for Σ_2SAT in the randomized setting and still hold the record, namely an almost-quadratic time lower bound for subpolynomial space.

Theorem 1.7 (Diehl–van Melkebeek [17]). For every real $d < 2$ there exists a positive real e such that Σ_2SAT cannot be solved by a randomized random-access machine with two-sided error that runs in time n^d and space n^e. Moreover, e approaches $1/2$ from below as d approaches 1 from above.

Note a few other differences with the deterministic setting. The format of Theorem 1.7 is weaker than that of Theorem 1.3, which entails machines of types (i) and (ii). In the randomized setting, we do not know how to take advantage of the existence of an algorithm for Σ_2SAT that runs in time n^c for small c but unrestricted space to derive better time–space lower bounds for Σ_2SAT. The parameters of Theorem 1.8 are also weaker than those of the corresponding result for Σ_2SAT in the deterministic setting, where the bound on d is larger than 2 and e converges to 1 when d goes to 1. See Section 4.2 for the exact bounds for Σ_2SAT in the deterministic setting.

Theorem 1.7 also applies to Π_2SAT, the complement of Σ_2SAT, as randomized computations with two-sided error can be trivially complemented. For the equivalents of satisfiability, tautologies, Σ_2SAT, Π_2SAT, etc. in higher levels of the polynomial-time hierarchy, stronger results can be shown, including results in the model where the randomized machines have two-way sequential access to the random-bit tape. Theorem 1.7 refers to the more natural but weaker coin flip model of space-bounded randomized computation, which can be viewed as equipping a deterministic machine with one-way access to a random bit tape. We refer to Section 7.2 for more details.

In the setting of one-sided error (with errors only allowed on the membership side), we do have lower bounds for the first level of the polynomial-time hierarchy, namely for tautologies. Such results trivially follow from Theorem 1.5 since randomized machines with one-sided error are special cases of nondeterministic machines. For example, we can conclude from Theorem 1.5 that tautologies cannot have both a randomized algorithm with one-sided error that runs in time $n^{1+o(1)}$ and a randomized algorithm with one-sided error that runs in polynomial time and subpolynomial space. Diehl and van Melkebeek [17] observed that the (then) known lower bound proofs for satisfiability on deterministic machines can be extended to lower bound proofs for tautologies on randomized machines with one-sided error without any loss in parameters. Their argument holds for all proofs to date, including Theorem 1.3. In particular, we know that tautologies cannot be solved by a randomized algorithm with one-sided error that runs in time $n^{2\cos(\pi/7)-o(1)}$ and subpolynomial space.

In the quantum setting, the simplest problem for which we currently know nontrivial lower bounds is MajMajSAT. MajSAT, short for majority-satisfiability, denotes the problem of deciding whether the majority of the assignments to a given Boolean formula satisfy the formula. Similarly, an instance of MajMajSAT asks whether a given Boolean formula depending on two sets of variables y and z has the property that for at least half of the assignments to y, at least half of the assignments to z satisfy the formula.

Allender et al. [3] showed a lower bound for MajMajSAT on randomized machines with unbounded error. The parameters are similar to those in Fortnow's time–space lower bound for satisfiability. In particular, they prove that MajMajSAT does not have a randomized algorithm with unbounded error that runs in time $n^{1+o(1)}$ and space $n^{1-\epsilon}$. van Melkebeek and Watson [41], building on earlier work by Adleman et al. [1], showed how to simulate quantum computations with bounded error on randomized machines with unbounded error in a time- and space-efficient way. As a result, they can translate the lower bound of Allender et al. to the quantum setting.

Theorem 1.8 (van Melkebeek–Watson [41], using Allender et al. [3]). For every real d and positive real ϵ there exists a real $c > 1$ such that at least one of the following fails:

(i) MajMajSAT has a quantum algorithm with two-sided error that runs in time n^c and
(ii) MajSAT has a quantum algorithm with two-sided error that runs in time n^d and space $n^{1-\epsilon}$.

Corollary 1.2. For every positive real ϵ there exists a real $d > 1$ such that MajMajSAT does not have a quantum algorithm with two-sided error that runs in time n^d and space $n^{1-\epsilon}$.

There is a — very simple — reduction from satisfiability to MajSAT but presumably not the other way around since MajSAT is hard for the entire polynomial-time hierarchy [54]. The same statement holds for MajMajSAT and Σ_2SAT instead of MajSAT and satisfiability, respectively. The reason why we have quantum lower bounds for MajMajSAT but not for Σ_kSAT for any integer k bears some similarity to why we have randomized lower bounds for Σ_2SAT but not for satisfiability. MajSAT tightly captures randomized computations with unbounded error in the same was as Σ_kSAT captures Σ_k-computations. We can efficiently simulate randomized computations with two-sided error on Σ_2-machines but we do not know how to do so on nondeterministic machines. Similarly, we can efficiently simulate quantum computations with bounded error on randomized machines with unbounded error but we do not know how to do that on Σ_k-machines. This analogy actually suggests that we ought to get quantum lower bounds for MajSAT rather than only for MajMajSAT. We discuss that prospect in Chapter 9.

1.1 Scope

This paper surveys the known robust lower bounds for the time and space complexity of satisfiability and closely related problems on general-purpose models of computation. The bounds depend on the fundamental capabilities of the model (deterministic, randomized, quantum, etc.) but are robust, up to polylogarithmic factors, with respect to the details of the model specification. For each of the basic models, we focus on the simplest problem for which we can establish nontrivial lower bounds. Except for the randomized and quantum models, that problem is satisfiability (or tautologies).

We do not cover lower bounds on restricted models of computation. The latter includes general-purpose models without random access, such as one-tape Turing machines with sequential access, off-line Turing machines (which have random access to the input and sequential access to a single work tape), and multi-tape Turing machines with sequential access via one or multiple tape heads. In those models, techniques from communication complexity can be used to derive lower bounds for simple problems like deciding palindromes or computing generalized inner products. Time–space lower bounds for such problems immediately imply time–space lower bounds for satisfiability by virtue of the very efficient reductions to satisfiability. However, in contrast to the results we cover, these arguments do not rely on the inherent difficulty of satisfiability. They rather exploit an artifact of the model of computation, e.g., that a one-tape Turing machine with sequential access deciding palindromes has to waste a lot of time in moving its tape head between both ends of the tape. Note that on random-access machines palindromes and generalized inner products can be computed simultaneously in quasi-linear time and logarithmic space. We point out that some of the techniques in this survey lead to improved results on some restricted models of computation, too, but we do not discuss them.

Except in Corollary 1.1, we also do not consider restricted circuit models. In several of those models lower bounds have been established for problems computable in polynomial time. Such results imply lower bounds for satisfiability on the same model provided the problems reduce to satisfiability in a simple way. As we will see in Section 2.3, problems in nondeterministic quasi-linear time are precisely those that have this property in a strong sense — they translate to satisfiability in quasi-linear time and do so in an oblivious way. All of the classical lower bounds on restricted circuit models involve problems in nondeterministic quasi-linear time and therefore also hold for satisfiability up to polylogarithmic factors. These results include the exponential lower bounds for the size of constant-depth circuits (for parity and its cousins), the quadratic lower bound for branching program size (for a version of the element distinctness problem, whose complement lies in nondeterministic quasi-linear time), and the cubic lower bound for formula size (for Andreev's addressing function). See [9] for a survey that is still up to date in terms of the strengths of the bounds except for the formula size lower bound [25]. We point out that some of the more recent work in circuit complexity does not seem to have implications for satisfiability. In particular, the non-uniform time–space lower bounds by Ajtai [2] and their improvements by Beame et al. [7] do not yield time–space lower bounds for satisfiability. These authors consider a problem in P based on a binary quadratic form, and showed that any branching program for it that uses only $n^{1-\epsilon}$ space for some positive constant ϵ takes time

$$\Omega(n \cdot \sqrt{\log n / \log \log n}). \tag{1.1}$$

An extremely efficient reduction of the problem they considered to satisfiability is needed in order to obtain nontrivial lower bounds for satisfiability, since the bound (1.1) is only slightly super-linear. The best known reductions (see Section 2.3.1) do not suffice. Moreover, their problem does not appear to be in nondeterministic quasi-linear time.

1.2 Organization

Chapter 2 contains preliminaries. Although the details of the model of computation do not matter, we describe a specific model for concreteness. We also specify our notation for complexity classes and exhibit complete problems which capture those classes very tightly such that time–space lower bounds for those problems and for linear time on the corresponding models are equivalent up to polylogarithmic factors. Whereas in this section we have stated all results in terms of the complete problems, in the rest of the paper we will think in terms of linear time on the corresponding models.

We present the known results in a unified way by distilling out what they have in common. Chapter 3 introduces the proof techniques and the tools involved in proving many of the lower bounds. It turns out that all the proofs have a very similar high-level structure, which can be characterized as indirect diagonalization. We describe how it works, what the ingredients are, and illustrate how they can be combined.

We then develop the results for the various models within this unifying framework: deterministic algorithms in Chapter 4, nondeterministic algorithms in Chapter 5, somewhat-nonuniform algorithms in Chapter 6, randomized algorithms in Chapter 7, and quantum algorithms in Chapter 8. We mainly focus on space bounds of the form $n^{1-\epsilon}$ and on subpolynomial space bounds as they allow us to present the underlying ideas without getting bogged down in notation and messy calculations. Chapters 4 through 8 are largely independent of each other, although some familiarity with the beginning of Chapter 4 can help to better appreciate Chapters 5, 6, and 7.

Finally, in Chapter 9 we propose some directions for further research.

References

[1] L. Adleman, J. DeMarrais, and M. Huang, “Quantum computability,” *SIAM Journal on Computing*, vol. 26, pp. 1524–1540, 1997.

[2] M. Ajtai, “A non-linear time lower bound for Boolean branching programs,” in *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*, pp. 60–70, IEEE, 1999.

[3] E. Allender, M. Koucky, D. Ronneburger, S. Roy, and V. Vinay, “Time-space tradeoffs in the counting hierarchy,” in *Proceedings of the 16th IEEE Conference on Computational Complexity*, pp. 295–302, IEEE, 2001.

[4] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy, “Proof verification and the hardness of approximation problems,” *Journal of the ACM*, vol. 45, no. 3, pp. 501–555, 1998.

[5] L. Babai, “Trading group theory for randomness,” in *Proceedings of the 17th ACM Symposium on the Theory of Computing*, pp. 421–429, ACM, 1985.

[6] L. Babai and S. Moran, “Arthur-Merlin games: A randomized proof system, and a hierarchy of complexity classes,” *Journal of Computer and System Sciences*, vol. 36, pp. 254–276, 1988.

[7] P. Beame, M. Saks, X. Sun, and E. Vee, “Time-space trade-off lower bounds for randomized computation of decision problems,” *Journal of the ACM*, vol. 50, no. 2, pp. 154–195, 2003.

[8] E. Bernstein and U. Vazirani, “Quantum complexity theory,” *SIAM Journal on Computing*, vol. 26, pp. 1411–1473, 1997.

[9] R. Boppana and M. Sipser, “Complexity of finite functions,” in *Handbook of Theoretical Computer Science*, (J. van Leeuwen, ed.), pp. 758–804, MIT Press, 1990.

[10] A. Chiu, *Complexity of Parallel Arithmetic Using The Chinese Remainder Representation.* Master's thesis, University of Wisconsin-Milwaukee, 1995.

[11] A. Chiu, G. Davida, and B. Litow, "Division in logspace-uniform NC^1," *Theoretical Informatics and Applications*, vol. 35, pp. 259–276, 2001.

[12] A. Cohen and A. Wigderson, "Dispersers, deterministic amplification, and weak random sources," in *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science*, pp. 14–19, IEEE, 1989.

[13] S. Cook, "A hierarchy theorem for nondeterministic time complexity," *Journal of Computer and System Sciences*, vol. 7, pp. 343–353, 1973.

[14] S. Cook, "Short propositional formulas represent nondeterministic computations," *Information Processing Letters*, vol. 26, pp. 269–270, 1988.

[15] T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms.* MIT Press, 2nd ed., 2001.

[16] S. Diehl, "Lower bounds for swapping Arthur and Merlin," in *Proceedings of the 11th International Workshop on Randomized Techniques in Computation*, pp. 449–463, Springer-Verlag, 2007.

[17] S. Diehl and D. van Melkebeek, "Time-space lower bounds for the polynomial-time hierarchy on randomized machines," *SIAM Journal on Computing*, vol. 36, pp. 563–594, 2006.

[18] S. Diehl, D. van Melkebeek, and R. Williams, "A new time-space lower bound for nondeterministic algorithms solving tautologies," Tech. Rep. 1601, Department of Computer Sciences, University of Wisconsin-Madison, 2007.

[19] L. Fortnow, "Time-space tradeoffs for satisfiability," *Journal of Computer and System Sciences*, vol. 60, pp. 337–353, 2000.

[20] L. Fortnow, R. Lipton, D. van Melkebeek, and A. Viglas, "Time-space lower bounds for satisfiability," *Journal of the ACM*, vol. 52, pp. 835–865, 2005.

[21] L. Fortnow and J. Rogers, "Complexity limitations on quantum computations," *Journal of Computer and System Sciences*, vol. 59, pp. 240–252, 1990.

[22] L. Fortnow and D. van Melkebeek, "Time-space tradeoffs for nondeterministic computation," in *Proceedings of the 15th IEEE Conference on Computational Complexity*, pp. 2–13, IEEE, 2000.

[23] O. Gabber and Z. Galil, "Explicit constructions of linear-sized superconcentrators," *Journal of Computer and System Sciences*, vol. 22, pp. 407–420, 1981.

[24] O. Goldreich and D. Zuckerman, "Another proof that $\mathrm{BPP} \subseteq \mathrm{PH}$ (and more)," Tech. Rep. TR-97-045, Electronic Colloquium on Computational Complexity, 1997.

[25] J. Hastad, "The shrinkage exponent of de Morgan formulas is 2," *SIAM Journal on Computing*, vol. 27, pp. 48–64, 1998.

[26] F. Hennie and R. Stearns, "Two-tape simulation of multitape Turing machines," *Journal of the ACM*, vol. 13, pp. 533–546, 1966.

[27] W. Hesse, "Division is in uniform TC^0," in *Proceedings of the 28th International Colloquium On Automata, Languages and Programming*, pp. 104–114, Springer-Verlag, 2001.

[28] W. Hesse, E. Allender, and D. M. Barrington, "Uniform constant-depth threshold circuits for division and iterated multiplication," *Journal of Computer and System Sciences*, vol. 65, pp. 695–712, 2002.

[29] R. Impagliazzo, N. Nisan, and A. Wigderson, "Pseudorandomness for network algorithms," in *Proceedings of the 26th ACM Symposium on the Theory of Computing*, pp. 356–364, ACM, 1994.

[30] R. Impagliazzo and D. Zuckerman, "How to recycle random bits," in *Proceedings of the 30th IEEE Symposium on Foundations of Computer Science*, pp. 248–253, IEEE, 1989.

[31] R. Kannan, "Towards separating nondeterminism from determinism," *Mathematical Systems Theory*, vol. 17, pp. 29–45, 1984.

[32] R. Kannan, H. Venkateswaran, V. Vinay, and A. Yao, "A circuit-based proof of Toda's theorem," *Information and Computation*, vol. 104, pp. 271–276, 1993.

[33] R. Karp and R. Lipton, "Turing machines that take advice," *L'Enseignement Mathématique*, vol. 28, no. 2, pp. 191–209, (A preliminary version appeared in STOC 1980), 1982.

[34] A. Kitaev, "Quantum computations: Algorithms and error correction," *Russian Mathematical Surveys*, vol. 52, pp. 1191–1249, 1997.

[35] A. Kitaev, A. Shen, and M. Vyalyi, *Classical and Quantum Computation*. American Mathematical Society, 2002.

[36] C. Lautemann, "BPP and the polynomial hierarchy," *Information Processing Letters*, vol. 17, pp. 215–217, 1983.

[37] R. Lipton and A. Viglas, "On the complexity of SAT," in *Proceedings of the 40th IEEE Symposium on Foundations of Computer Science*, pp. 459–464, IEEE, 1999.

[38] G. Margulis, "Explicit construction of concentrators," *Problems of Information Transmission*, vol. 9, pp. 325–332, 1973.

[39] D. van Melkebeek, "Time-space lower bounds for NP-complete problems," in *Current Trends in Theoretical Computer Science*, (G. Paun, G. Rozenberg, and A. Salomaa, eds.), pp. 265–291, World Scientific, 2004.

[40] D. van Melkebeek and K. Pervyshev, "A generic time hierarchy for semantic models with one bit of advice," *Computational Complexity*, vol. 16, pp. 139–179, 2007.

[41] D. van Melkebeek and T. Watson, "A quantum time-space lower bound for the counting hierarchy," Tech. Rep. 1600, Department of Computer Sciences, University of Wisconsin-Madison, 2007.

[42] A. Naik, K. Regan, and D. Sivakumar, "On quasilinear-time complexity theory," *Theoretical Computer Science*, vol. 148, pp. 325–349, 1995.

[43] V. Nepomnjascii, "Rudimentary predicates and Turing calculations," *Soviet Mathematics–Doklady*, vol. 11, pp. 1462–1465, 1970.

[44] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[45] N. Nisan, "Pseudorandom generators for space-bounded computation," *Combinatorica*, vol. 12, pp. 449–461, 1992.

[46] N. Nisan, "$\mathrm{RL} \subseteq \mathrm{SC}$," *Computational Complexity*, vol. 4, pp. 1–11, 1994.

[47] N. Nisan and A. Wigderson, "Hardness vs. randomness," *Journal of Computer and System Sciences*, vol. 49, pp. 149–167, 1994.

[48] N. Pippenger and M. Fischer, "Relations among complexity measures," *Journal of the ACM*, vol. 26, pp. 361–381, 1979.

[49] J. Robson, "An $O(T \log T)$ reduction from RAM computations to satisfiability," *Theoretical Computer Science*, vol. 82, pp. 141–149, 1991.

[50] W. Ruzzo, "Tree-size bounded alternation," *Journal of Computer and System Sciences*, vol. 21, pp. 218–235, 1980.

[51] W. Savitch, "Relationships between nondeterministic and deterministic tape complexities," *Journal of Computer and System Sciences*, vol. 4, pp. 177–192, 1970.

[52] J. Seiferas, M. Fischer, and A. Meyer, "Separating nondeterministic time complexity classes," *Journal of the ACM*, vol. 25, pp. 146–167, 1978.

[53] M. Sipser, "A complexity theoretic approach to randomness," in *Proceedings of the 15th ACM Symposium on the Theory of Computing*, pp. 330–335, ACM, 1983.

[54] S. Toda, "PP is as hard as the polynomial-time hierarchy," *SIAM Journal on Computing*, vol. 20, no. 5, pp. 865–877, 1991.

[55] J. Toran, "Complexity classes defined by counting quantifiers," *Journal of the ACM*, vol. 38, pp. 753–774, 1991.

[56] I. Tourlakis, "Time-space lower bounds for SAT on nonuniform machines," *Journal of Computer and System Sciences*, vol. 63, no. 2, pp. 268–287, 2001.

[57] L. Valiant and V. Vazirani, "NP is as easy as detecting unique solutions," *Theoretical Computer Science*, vol. 47, pp. 85–93, 1986.

[58] E. Viola, "On approximate majority and probabilistic time," in *Proceedings of the 22nd IEEE Conference on Computational Complexity*, pp. 155–168, IEEE, 2007.

[59] R. Williams, "Better time-space lower bounds for SAT and related problems," in *Proceedings of the 20th IEEE Conference on Computational Complexity*, pp. 40–49, IEEE, 2005.

[60] R. Williams, *Algorithms and resource requirements for fundamental problems.* PhD thesis, Carnegie Mellon Univesity, 2007.

[61] R. Williams, "Time-space tradeoffs for counting NP solutions modulo integers," in *Proceedings of the 22nd IEEE Conference on Computational Complexity*, pp. 70–82, IEEE, 2007.

[62] C. Yap, "Some consequences of non-uniform conditions on uniform classes," *Theoretical Computer Science*, vol. 26, pp. 287–300, 1983.

[63] S. Zak, "A Turing machine time hierarchy," *Theoretical Computer Science*, vol. 26, pp. 327–333, 1983.