Circuit Bottom Fan-in and Computational Power

LIMING CAI[*]
Department of Mathematics
East Carolina University
Greenville, NC 27858-4353
cai@math1.math.ecu.edu

JIANER CHEN[†]
Department of Computer Science
Texas A&M University
College Station, TX 77843-3112
chen@cs.tamu.edu

JOHAN HÅSTAD
Department of Computer Science
Royal Institute of Technology
Stockholm, Sweden
johanh@nada.kth.se

Abstract

We investigate the relationship between circuit bottom fan-in and circuit size when circuit depth is fixed. We show that in order to compute certain functions, a moderate reduction in circuit bottom fan-in will cause significant increase in circuit size. In particular, we prove that there are functions that are computable by circuits of linear size and depth k with bottom fan-in 2 but require exponential size for circuits of depth k with bottom fan-in 1. A general scheme is established to study the trade-off between circuit bottom fan-in and circuit size. Based on this scheme, we are able to prove, for example, that for any integer c, there are functions that are computable by circuits of linear size and depth k with bottom fan-in $O(\log n)$ but require exponential size for circuits of depth k with bottom fan-in c, and that for any constant $\epsilon > 0$, there are functions that are computable by circuits of linear size and depth k with bottom fan-in $\log n$ but require super-polynomial size for circuits of depth k with bottom fan-in $O(\log^{1-\epsilon} n)$. A consequence of these results is that the three input read-modes of alternating Turing machines proposed in the literature are all distinct.

[*] Supported in part by Engineering Excellence Award from Texas A&M University.

[†] Supported in part by the National Science Foundation under Grant CCR-9110824.

Key words:

computational complexity, circuit complexity, lower bound, alternating Turing machine

AMS(MOS) subject classification:

68Q05, 68Q10, 68Q15, 68Q25, 68Q30

Abbreviated title:

Circuit Bottom Fan-in

The Corresponding Author

Jianer Chen
Department of Computer Science
Texas A&M University
College Station, TX 77843-3112
chen@cs.tamu.edu

1 Introduction

To prove lower bounds for various computational models remains as one of the most challenging tasks in complexity theory. Much progress has been made recently in deriving lower bounds for computational models with limited capabilities, with the hope that these may lead to better lower bounds for more general computational models and to better understanding of intrinsic complexity of computation.

One of the most successful trials is the derivation of lower bounds for constant depth circuits. The first strong lower bounds were given by Furst, Saxe, and Sipser [12], independently by Ajtai [1], who show that the size of a constant depth circuit computing the parity function is superpolynomial. The results were subsequently sharpened by Yao [18] who derived an exponential lower bound. Håstad [14, 15] further strengthened the result and obtained near optimal lower bounds. A direct consequence of these results is that the logarithmic time hierarchy [17], i.e., the set of languages accepted by families of circuits of constant depth and polynomial size, is a proper subset of P.

The logarithmic time hierarchy was further refined by Sipser [17] who showed that for each integer $k > 1$, there are functions that are computable by a circuit of depth k and polynomial size but require superpolynomial size for circuits of depth $k - 1$. Thus, all levels of the logarithmic time hierarchy are distinct. Exponential lower bounds for the depth k to $k - 1$ conversion were first claimed by Yao [18] and then fully proved by Håstad [14, 15].

In this paper, we will further sharpen the separation results in the logarithmic time hierarchy by investigating the relationship between circuit bottom fan-in and circuit size when circuit depth is fixed. We show that in order to compute certain functions, a moderate reduction in circuit bottom fan-in will cause significant increase in circuit size. In particular, we prove that there are functions that are computable by circuits of linear size and depth k with bottom fan-in 2 but require exponential size for circuits of depth k with bottom fan-in 1. A general scheme is established to study the trade-off between circuit bottom fan-in and circuit size. Based on this scheme, we are able to prove, for example, that for any integer c, there are functions that are computable by circuits of linear size and depth k with bottom fan-in $O(\log n)$ but require exponential size for circuits of depth k with bottom fan-in c, and that for any constant $\epsilon > 0$, there are functions that are computable by circuits of linear size and depth k with bottom fan-in $\log n$ but require superpolynomial size for circuits of depth k with bottom fan-in $O(\log^{1-\epsilon} n)$. Therefore, the computational power of constant depth circuits depends not only on its depth, but also strictly on its bottom fan-in when the depth of the circuits is fixed.

Another motivation of our present research is from the study of input read-modes of a sublinear-time alternating Turing machine, which is an important computational model in the study of complexity classes. A number of input read-modes for sublinear-time alternating Turing machines have appeared in the literature. In the standard model proposed by Chandra, Kozen, and Stockmeyer [9], a computation path of the machine can read up to $O(\log n)$ input bits in time $O(\log n)$. Ruzzo [16] proposed an input read-mode in which each computation path can read at most one input bit and the reading must be performed at the end of the path. An input read-mode studied by Sipser [17] insists that each input reading takes time $\Omega(\log n)$. These input read-modes have been carefully studied by Cai and Chen [6], who have given a precise circuit characterization for each read-mode for log-time alternating Turing machines of constant alternations. Input read-modes of log-time alternating Turing machines also find applications in the study of computational optimization problems [5, 8] and in the study of limited nondeterminism [7].

Based on Cai and Chen's circuit characterizations and our separation results in constant depth circuits, we are able to show that the three proposed input read-modes for alternating Turing machines are all distinct. More precisely, if we let $\Pi_k^{\mathcal{U}}$ (resp. $\Pi_k^{\mathcal{R}}$, $\Pi_k^{\mathcal{S}}$) be the class of languages accepted by log-time k-alternation alternating Turing machines using Chandra, Kozen, and Stockmeyer's (resp. Ruzzo's, Sipser's) input read-mode, then we can show that for all integers $k \geq 1$

$$\Pi_k^{\mathcal{R}} \subset \Pi_k^{\mathcal{S}} \subset \Pi_k^{\mathcal{U}} \subset \Pi_{k+1}^{\mathcal{R}}$$

where $\subset$ means "proper subset". This gives a very detailed refinement of the logarithmic time hierarchy, and shows the rich structural properties of the logarithmic time hierarchy.

The paper is organized as follows. Section 2 introduces necessary definitions and related previous work. In Section 3, we show that in order to compute certain functions, an $O(\log n)$ factor reduction in circuit bottom fan-in may cause exponential increase in circuit size. In Section 4, we show that for circuits computing certain special functions, even reducing the circuit bottom fan-in by 1 will result in exponential increase in circuit size. A general scheme is established in Section 5 to study the trade-off between circuit bottom fan-in and circuit size. The relationship to the input read-modes of alternating Turing machines is given in Section 6.

2 Preliminaries

We briefly review the fundamentals related to the present paper. For further discussion on the theory of circuit complexity and alternating Turing machines, the reader is referred to [3, 11].

An (unbounded fan-in) *Boolean circuit* α_n with input $x = x_1 x_2 \cdots x_n$ of length n is a directed acyclic graph. The *fan-in* of a node in the circuit is the in-degree of the node. The nodes of fan-in 0 are called *inputs* and are labeled from the set $\{0, 1, x_1, \overline{x}_1, \cdots, x_n, \overline{x}_n\}$. The nodes of fan-in greater than 0 are called *gates* and are labeled either AND or OR. One of the nodes is designated the *output* node. The *size* is the number of gates, and the *depth* is the maximum distance from an input to the output. Without loss of generality, we assume the circuits are of the special form where all AND and OR gates are organized into alternating levels with edges only between adjacent levels. Any circuit may be converted to one of this form without increasing the depth and by at most squaring the size [10]. In this special form, the gates that are connected to input nodes will be called *bottom level gates*, or *depth* 1 *gates*. The gates that receive inputs ¿from depth 1 gates will be called *depth* 2 *gates*, and so on. The *bottom fan-in* of a circuit is the maximum over fan-ins of all bottom level gates. The following notation introduced by Boppana and Sipser [3] will be especially convenient in our discussion.

Definition [3] A circuit α is a Π_k^s*-circuit* (resp. Σ_k^s*-circuit*) if α is a depth k circuit of size at most s with an AND-gate (resp. an OR-gate) at the output. A circuit β is a $\Pi_k^{s,c}$*-circuit* (resp. $\Sigma_k^{s,c}$*-circuit*) if β is a depth $k+1$ circuit of size at most s with bottom fan-in c and an AND-gate (resp. an OR-gate) at the output.

A *family* of circuits is a sequence $\{\alpha_n \,|\, n \geq 1\}$ of circuits, where α_n is with input of length n. A family of circuits may be used to define a language. A family $\{\alpha_n \mid n \geq 1\}$ of circuits is said to be a Π_k^{poly}-family (resp. $\Pi_k^{poly,c}$-family) if there is a polynomial p such that for all $n \geq 1$, α_n is a $\Pi_k^{p(n)}$-circuit (resp. $\Pi_k^{p(n),c}$-circuit).

The *Sipser function* f_k^m, as defined in [14, 15], is given by the circuit

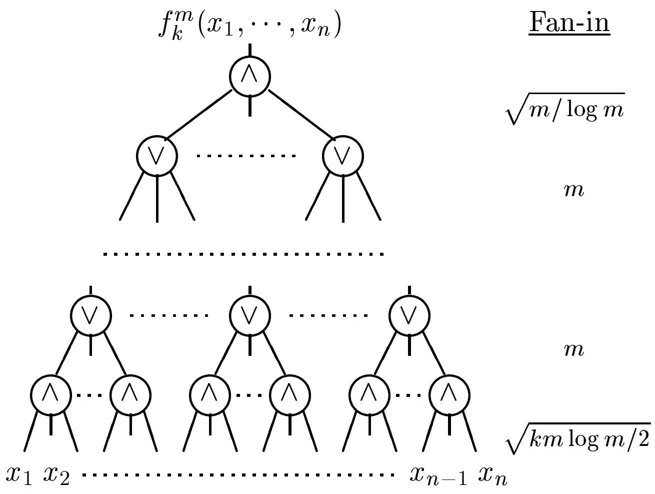

Figure 1: The circuit C_k^m defining the function f_k^m

C_k^m shown in Figure 1. The circuit C_k^m is a tree of depth k in which every gate in the bottom level has fan-in $\sqrt{km \log m/2}$, the fan-in of the output gate is $\sqrt{m/\log m}$, and the fan-in for all other gates is m. Each variable x_i, $1 \leq i \leq n$, occurs at only one leaf. Note that the number n of variables of the function f_k^m equals $m^{k-1}\sqrt{k/2}$.

The following theorem is proved by Håstad [14, 15].

Theorem 2.1 ([14, 15]) *There is no depth k circuit computing the function f_k^m with bottom fan-in $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$ and fewer than $2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$ gates of depth ≥ 2, for $m > m_0$, where m_0 is a absolute constant.*

To further sharpen the separation results and study the relationship between circuit bottom fan-in and circuit size, we introduce a variation $f_k^{m,b}$ of the Sipser function by explicitly specifying the bottom fan-in for the defining circuits.

Definition Let $C_k^{m,b}$ be the tree circuit defining the Sipser function f_k^m, as illustrated in Figure 1, except that each bottom level gate of $C_k^{m,b}$ has fan-in b instead of $\sqrt{km \log m/2}$. Define $f_k^{m,b}$ to be the function computed by the tree circuit $C_k^{m,b}$. The number n of variables of the function $f_k^{m,b}$ is

$n = bm^{k-2}\sqrt{m/\log m}$.

The discussion of the present paper is centered on the complexity of the function $f_k^{m,b}$.

3 On circuits that compute $f_k^{m,\Omega(\log m)}$

In this section, we consider the complexity of the function $f_k^{m,b}$, where b is of order $\Omega(\log m)$.

Our main result in this section is that for $b > k\log m$, the function $f_k^{m,b}$ cannot be computed by any depth k circuit without exponential size and with bottom fan-in $\frac{b}{c\log m}$ for a particular constant c. This result requires two different proofs depending on whether the bottom fan-in b is larger or smaller than the bottom fan-in of the standard Sipser function circuit C_k^m given in Figure 1.

We first consider the case $b \leq \sqrt{km\log m/2}$. For this, we need to briefly review some notation and results by Håstad [14].

Definition ([14]) Let q be a real number and $(B_i)_{i=1}^r$ a partition of the variables. Let $R_{q,B}^+$ be the probability space of restrictions ρ that take values as follows.

For every B_i, $1 \leq i \leq r$ independently

1. With probability q let $s_i = *$ and else $s_i = 0$.
2. For every $x_k \in B_i$ let $\rho(x_k) = s_i$ with probability q and else $\rho(x_k) = 1$.

Similarly, an $R_{q,B}^-$ probability space of restrictions is defined by interchanging the roles played by 0 and 1.

The idea behind these restrictions is that a block B_i will correspond to the variables leading into one of the bottom level gates of the circuit $C_k^{m,b}$ that defines the function $f_k^{m,b}$. If the bottom level gates of $C_k^{m,b}$ are ANDs, we use a restriction from $R_{q,B}^+$; if the bottom level gates of $C_k^{m,b}$ are ORs, we use a restriction from $R_{q,B}^-$.

Definition ([14]) For a restriction $\rho \in R_{q,B}^+$, let $g(\rho)$ be the restriction defined as follows: For all B_i with $s_i = *$, $g(\rho)$ gives the value 1 to all variables which are given value $*$ by ρ except the one with the highest index among those variables given value $*$ by ρ, to which $g(\rho)$ gives value $*$.

If $\rho \in R^{-}_{q,B}$, then $g(\rho)$ is defined similarly but now takes the value 0 and $*$. Note that for a given restriction ρ, the restriction $g(\rho)$ can be obtained by a deterministic process that makes each block B_i have at most one $*$.

Let $\rho g(\rho)$ denote the composition of the two restrictions. That is, $\rho g(\rho)$ is the restriction $g(\rho)$ obtained from the restriction ρ.

Lemma 3.1 (The Switching Lemma [14]) *Let σ be an* AND *of* OR*s all of size $\leq t$ and $\rho \in R^{+}_{q,B}$. Then the probability that under the restriction $\rho g(\rho)$ σ cannot be written as an* OR *of* AND*s all of size $< s$ is bounded by α^s, where $\alpha < \frac{4qt}{\ln 2} < 5.78qt$.*

The Switching Lemma is also true if we do either or both of the following replacements: (1) replacing the probability space $R^{+}_{q,B}$ by the probability space $R^{-}_{q,B}$; and (2) replacing σ by an OR of ANDs to be converted to an AND and ORs.

Now we are ready for our first main result of this section.

Theorem 3.2 *For $k \log m < b \leq \sqrt{km \log m/2}$, the function $f_k^{m,b}$ cannot be computed by any depth k circuit of bottom fan-in $\frac{b}{12k \log m}$ and size bounded by $2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$, for $m > m_0$, where m_0 is an absolute constant.*

Proof. The proof is similar to the induction step in the proof given by Håstad for Theorem 2.1 (see [14], pages 48-50). Therefore, we only outline the proof and describe in detail those places that are different.

We set $q = \frac{k \log m}{b}$. Suppose that $\tau_1, \ldots, \tau_r$, $r = m^{k-2}\sqrt{m/\log m}$, are the bottom level gates of the tree circuit $C_k^{m,b}$ defining the function $f_k^{m,b}$. Let $(B_j)_{j=1}^r$ be the partition of the variables of $f_k^{m,b}$ such that block B_j is the set of variables leading into the bottom level gate τ_j of $C_k^{m,b}$.

Claim 1: The probability that under the restriction $\rho g(\rho)$ any bottom level gate τ_j of the tree circuit $C_k^{m,b}$ does not take value s_j is bounded by $\frac{1}{6m}$.

The proof is identical to the proof by Håstad for Fact 1 in [14] (page 49): Such a gate τ_j does not take the corresponding value s_j with probability $(1-q)^{|B_j|} < \frac{1}{6}m^{-k}$. Since there are fewer than m^{k-1} bottom level gates in the tree circuit $C_k^{m,b}$, the probability in Claim 1 is bounded by $\frac{1}{6m}$.

Claim 2: The probability that under the restriction $\rho g(\rho)$ any depth 2 gate in the tree circuit $C_k^{m,b}$ gets fewer than $\sqrt{(k-1)m\log m/2}$ $*$'s from the bottom level is bounded by $\frac{1}{m}$.

Let $p_i = \binom{m}{i}q^i(1-q)^{m-i}$ be the probability that a fixed depth 2 gate μ in the tree circuit $C_k^{m,b}$ gets exactly i $*$'s from the bottom level. With the condition $b \le \sqrt{km\log m/2}$, we can show that for $i \le \sqrt{(k-1)m\log m}$, we have $\frac{p_i}{p_{i-1}} \ge \sqrt{2}$. Thus the probability that the gate μ gets fewer than $\sqrt{(k-1)m\log m/2}$ $*$'s is bounded by m^{-k} for sufficiently large m. Since there are fewer than m^{k-1} depth 2 gates in the tree circuit $C_k^{m,b}$, the probability in Claim 2 is bounded by $\frac{1}{m}$.

Now suppose that the theorem is not true. Thus, there is a depth k circuit C_0 of size bounded by $2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$ and bottom fan-in $t \le \frac{b}{12k\log m}$ that computes the function $f_k^{m,b}$. Furthermore, assume that the gates in the bottom level of the circuit C_0 are OR gates (the dual case can be proved similarly).

Claim 3: The probability that under the restriction $\rho g(\rho)$ any depth 2 gate in the circuit C_0 cannot be written as an OR of ANDs of size $\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}$ is bounded by $\frac{1}{2}$.

Let σ be a fixed depth 2 gate in the circuit C_0. By the Switching Lemma, under the restriction $\rho g(\rho)$, the probability that σ cannot be written as an OR of ANDs of size $\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}$ is bounded by $(5.78qt)^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$. Since the circuit C_0 has at most $2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$ depth 2 gates, the probability in Claim 3 is bounded by

$$\begin{array}{rc} & (5.78qt)^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}} \cdot 2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}} \\ < & (\frac{5.78}{12})^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}} \cdot 2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}} \\ \le & (\frac{5.78}{6})^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}} \end{array}$$

which is smaller than $\frac{1}{2}$ when m is sufficiently large.

Therefore with a probability larger than $1 - (\frac{1}{6m} + \frac{1}{m} + \frac{1}{2}) > \frac{1}{3}$, the tree circuit $C_k^{m,b}$ becomes a circuit that computes a function f at least as hard as the Sipser function f_{k-1}^m, and the circuit C_0 becomes a depth $k-1$ circuit that computes the function f and has bottom fan-in $\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}$ and fewer than $2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$ gates of depth ≥ 2. But this contradicts Theorem 2.1. $\square$

Letting $b = \sqrt{km \log m/2}$ in Theorem 3.2, we obtain Theorem 2.1. Note that the size bound is slightly improved.

Note that the condition $b \leq \sqrt{km \log m/2}$ in Theorem 3.2 is essential in the proof for Claim 2. For larger bottom fan-in b, we have the following theorem.

Theorem 3.3 *For $b \geq 2\sqrt{km \log m/2}$, the function $f_k^{m,b}$ cannot be computed by any depth k circuit of bottom fan-in $\frac{b}{25ke \log m}$ and size bounded by $2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$, for $m > m_0$, where m_0 is an absolute constant and e is the base of the natural logarithm.*

Proof. Let $q = \frac{1.04\sqrt{km \log m/2}}{b}$. Consider the following probability space R_q^+ of restrictions:

> For each variable x_k of the function $f_k^{m,b}$, let $\rho^+(x_k) = *$ with probability q and else $\rho^+(x_k) = 1$.

The probability space R_q^- is defined similarly except that the value 1 is replaced by value 0.

¿From now on, we assume that the bottom level gates of the tree circuit $C_k^{m,b}$ defining $f_k^{m,b}$ are AND gates. The case when the bottom level gates of $C_k^{m,b}$ are OR gates can be proved similarly by using the probability space R_q^- instead of the probability space R_q^+.

We first show that under a restriction $\rho^+ \in R_q^+$, with very large probability, the tree circuit $C_k^{m,b}$ computes a function at least as hard as the Sipser function f_k^m. The proof is similar to that for Claim 2 in Theorem 3.2. Thus, we only describe the differences.

Let τ be a bottom level gate in the tree circuit $C_k^{m,b}$. The gate τ is an AND gate of fan-in b. Let $p_i = \binom{b}{i} q^i (1-q)^{b-i}$ be the probability that the gate τ gets exactly i $*$'s under a restriction $\rho^+ \in R_q^+$. First we consider the ratio

$$\frac{p_i}{p_{i-1}} = \frac{b-i+1}{i} \cdot \frac{q}{1-q} > \frac{b-i}{i} \cdot \frac{q}{1-q}$$

For $i \leq 1.02\sqrt{km \log m/2}$, we have

$$\frac{b-i}{i} \geq \frac{b - 1.02\sqrt{km \log m/2}}{1.02\sqrt{km \log m/2}}$$

and

$$\frac{q}{1-q} = \frac{(1.04\sqrt{km\log m/2})/b}{1-(1.04\sqrt{km\log m/2})/b} = \frac{1.04\sqrt{km\log m/2}}{b-1.04\sqrt{km\log m/2}}$$

Thus, we have

$$\frac{p_i}{p_{i-1}} > \frac{b-1.02\sqrt{km\log m/2}}{1.02\sqrt{km\log m/2}} \cdot \frac{1.04\sqrt{km\log m/2}}{b-1.04\sqrt{km\log m/2}} \geq \frac{52}{51}$$

This gives $(51/52)^{j-i}p_j > p_i$ for $i < j \leq 1.02\sqrt{km\log m/2}$.

Now under a restriction $\rho^+ \in R_q^+$, the probability that the gate τ gets fewer than $\sqrt{km\log m/2}$ $*$'s is bounded by

$$\begin{array}{ccc} & \sum_{i=0}^{\sqrt{km\log m/2}} p_i < \sum_{i=0}^{\sqrt{km\log m/2}} (51/52)^{\sqrt{km\log m/2}-i} p_{\sqrt{km\log m/2}} \leq \\ \leq & 52 p_{\sqrt{km\log m/2}} < 52(51/52)^{0.02\sqrt{km\log m/2}} p_{1.02\sqrt{km\log m/2}} \leq \\ \leq & 52(51/52)^{0.02\sqrt{km\log m/2}} \end{array}$$

and $52(51/52)^{0.02\sqrt{km\log m/2}}$ is smaller than $\frac{1}{m^k}$ for sufficiently large m.

Since the circuit $C_k^{m,b}$ has fewer than m^{k-1} bottom level gates, we conclude that under a restriction $\rho^+ \in R_q^+$, the probability that any bottom level gate of the tree circuit $C_k^{m,b}$ gets fewer than $\sqrt{km\log m/2}$ $*$'s is bounded by $\frac{1}{m}$.

Now suppose that the theorem is not true. Thus, there is a depth k circuit C_0 of bottom fan-in at most $\frac{b}{25ke\log m}$ and size bounded by $2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$ such that the circuit C_0 computes the function $f_k^{m,b}$. We show that under a restriction $\rho^+ \in R_q^+$, with very large probability, the circuit C_0 becomes a depth k circuit of bottom fan-in at most $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$.

Let μ be a bottom level gate of fan-in $c \leq \frac{b}{25ke\log m}$ in the circuit C_0, and let $r_i = \binom{c}{i}q^i(1-q)^{c-i}$ be the probability that the gate μ gets exactly i $*$'s under a restriction $\rho^+ \in R_q^+$. We have

$$\binom{c}{i}q^i(1-q)^{c-i} \leq \frac{c!}{i!(c-i)!}q^i \leq \frac{c^iq^i}{i!} \leq \left(\frac{cqe}{i}\right)^i$$

where the last inequality is based on Stirling's approximation [13]

$$i! \geq 0.9(i/e)^i\sqrt{2\pi i} \geq (i/e)^i, \qquad \text{for } i \geq 1$$

Let $s = \frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$. Under a restriction $\rho^+ \in R_q^+$ the probability that the gate μ gets more than s $*$'s is bounded by

$$\sum_{i=s+1}^{c} r_i \leq \sum_{i=s+1}^{c} \left(\frac{cqe}{i}\right)^i \leq \sum_{i=s+1}^{c} \left(\frac{\frac{1.04}{25\sqrt{2k}}\sqrt{\frac{m}{\log m}}}{i}\right)^i$$

For $i > s = \frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$, we have $(\frac{1.04}{25\sqrt{2k}}\sqrt{\frac{m}{\log m}})/i < \frac{12.48}{25}$. Thus under a restriction $\rho^+ \in R_q^+$ the probability that the gate μ gets more than s $*$'s is bounded by

$$\sum_{i=s+1}^{c} (12.48/25)^i < (12.48/25)^s$$

Since the circuit C_0 has at most 2^s bottom level gates, we conclude that under a restriction $\rho^+ \in R_q^+$, the probability that any bottom level gate of the circuit C_0 gets more than s $*$'s is bounded by

$$(12.48/25)^s \cdot 2^s = (24.96/25)^s = (24.96/25)^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$$

which is smaller than $\frac{1}{m}$ for sufficiently large m.

Thus, under a restriction $\rho^+ \in R_q^+$, with probability $\geq 1 - \frac{1}{m} - \frac{1}{m} > \frac{1}{2}$, all bottom level gates of the tree circuit $C_k^{m,b}$ get at least $\sqrt{km \log m/2}$ $*$'s (thus $C_k^{m,b}$ is converted to a circuit computing a function at least as hard as the Sipser function f_k^m), and all bottom level gates of the circuit C_0 get at most $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$ $*$'s. Note that if a bottom level gate μ of the circuit C_0 gets at most $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$ $*$'s, then either the gate μ is eliminated from the bottom level (e.g., μ is an AND gate and gets an input with value 0) or the gate μ becomes a gate of fan-in at most $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$. In any case, we have derived that there is an assignment that converts the circuit C_0 into a depth k circuit C' of bottom fan-in bounded by $\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}$ and size bounded by $2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$ such that the circuit C' computes a function at least as hard as the Sipser function f_k^m. But this contradicts Theorem 2.1.

This completes the proof. □

4 On circuits that compute $f_k^{m,2}$

In the previous section, we showed that for circuits to compute the function $f_k^{m,b}$, $b = \Omega(\log m)$, an $O(\log m)$ factor reduction in bottom fan-in may cause an exponential increase in the circuit size. In this section, we will show that in certain cases, even reducing the circuit bottom fan-in by 1 will cause an exponential increase in the circuit size. More precisely, we will show that the function $f_k^{m,2}$ can be computed by a depth k circuit of linear size and bottom fan-in 2, but requires exponential size for depth k circuits of bottom fan-in 1. Note that a depth k circuit of bottom fan-in 1 is actually a depth $k-1$ circuit.

We prove the above result with a new probability space of restrictions. We start with the following lemma.

Lemma 4.1 *Partition the Boolean variables $\{x_1, \cdots, x_n\}$ into groups of c variables each. For each group, randomly pick r variables and assign them 0, and assign the rest $c-r$ variables $*$. Let σ be an* OR *of a subset S_σ of $\{x_1, \overline{x}_1, \cdots, x_n, \overline{x}_n\}$ such that S_σ contains at least h negative literals $\overline{x}_i$. Then with the above random assignment,*

$$Pr[\sigma \not\equiv 1] \leq ((c-r)/c)^h$$

PROOF. Let $s = n/c$ be the number of groups in the partition given in the statement of the lemma. We first rename the literals in the set S_σ so that $\overline{x}_1^{(d)}$, ..., $\overline{x}_{j_d}^{(d)}$, $d = 1, \ldots, s$, $j_1 + \cdots + j_s = h$, are h negative literals in S_σ, where $\overline{x}_1^{(d)}$, ..., $\overline{x}_{j_d}^{(d)}$ belong to the same group in the partition, $d = 1, \ldots, s$.

Let $A_t^{(d)}$ be the event that the variable $x_t^{(d)}$ is *not* assigned 0 by the random assignment, and let $\mathcal{E}_{\sigma\not\equiv 1}$ be the event that σ is not identical to 1. Then

$$\mathcal{E}_{\sigma\not\equiv 1} \subseteq \bigcap_{d=1}^{s} (A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)})$$

Thus

$$Pr[\sigma \not\equiv 1] = Pr[\mathcal{E}_{\sigma\not\equiv 1}] \leq Pr[\bigcap_{d=1}^{s} (A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)})] = \prod_{d=1}^{s} Pr[A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)}]$$

The last equality is because the event $A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)}$ and the event $A_1^{(d')} \cap \cdots \cap A_{j_{d'}}^{(d')}$ are independent for $d \neq d'$.

Now consider $Pr[A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)}]$. If $j_d > c - r$, then by the way we assign the dth group, at least one of the variables $x_1^{(d)}$, ..., $x_{j_d}^{(d)}$ is assigned 0. Therefore,

$$Pr[A_1^{(d)} \cap A_2^{(d)} \cap \cdots \cap A_{j_d}^{(d)}] = 0$$

If $j_d \leq c - r$, then

$$Pr[A_1^{(d)} \cap A_2^{(d)} \cap \cdots \cap A_{j_d}^{(d)}] =$$
$$Pr[A_1^{(d)}] \cdot Pr[A_2^{(d)} | A_1^{(d)}] \cdot Pr[A_3^{(d)} | A_1^{(d)} \cap A_2^{(d)}] \cdots Pr[A_{j_d}^{(d)} | A_1^{(d)} \cap \cdots \cap A_{j_d - 1}^{(d)}]$$

Note that

$$Pr[A_i^{(d)} | A_1^{(d)} \cap \cdots \cap A_{i-1}^{(d)}] = \frac{c - r - i + 1}{c - i + 1} \leq \frac{c - r}{c}$$

Thus,

$$Pr[A_1^{(d)} \cap A_2^{(d)} \cap \cdots \cap A_{j_d}^{(d)}] \leq ((c - r)/c)^{j_d}$$

This gives directly

$$Pr[\sigma \not\equiv 1] \leq \prod_{d=1}^{s} Pr[A_1^{(d)} \cap \cdots \cap A_{j_d}^{(d)}] \leq \prod_{d=1}^{s} \left(\frac{c - r}{c} \right)^{j_d} = \left(\frac{c - r}{c} \right)^{h}$$

□

Similarly we can prove

Lemma 4.2 *Partition the Boolean variables $\{x_1, \cdots, x_n\}$ into groups of c variables each. For each group, randomly pick r variables and assign them 1, and assign the rest $c - r$ variables $*$. Let σ be an* OR *of a subset S_σ of $\{x_1, \overline{x}_1, \cdots, x_n, \overline{x}_n\}$ such that S_σ contains at least h positive literals x_i. Then with the above random assignment,*

$$Pr[\sigma \not\equiv 1] \leq ((c - r)/c)^h$$

Now we are ready for the main theorem of this section.

Theorem 4.3 *The function $f_k^{m,2}$ cannot be computed by any depth $k - 1$ circuit of size bounded by $2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$ for $m > m_0$, where m_0 is an absolute constant.*

Proof. To simplify the expressions, we let $s = \frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}$. Suppose that the theorem is not true and that there is a depth $k - 1$ circuit C of size 2^s that computes the function $f_k^{m,2}$. Furthermore, we assume that the gates in the bottom level of C are OR gates (the case that the bottom level gates of C are AND gates can be proved similarly.)

Randomly pick one variable from each pair x_{2i-1} and x_{2i} and assign it 0. This will reduce the tree circuit $C_k^{m,2}$ defining $f_k^{m,2}$ to the tree circuit $C_{k-1}^{m,m}$ defining $f_{k-1}^{m,m}$.

Let τ be an OR gate in the bottom level of the circuit C such that τ has more than s negative literals in its input, then by Lemma 4.1,

$$Pr[\tau \not\equiv 1] \leq (1/2)^{s+1}$$

Let $\tau_1, \cdots, \tau_r$, $r \leq 2^s$, be all the gates in the bottom level of the circuit C such that there are more than s negative literals in their input, then

$$Pr[\tau_1 \not\equiv 1 \vee \cdots \vee \tau_r \not\equiv 1] \leq Pr[\tau_1 \not\equiv 1] + \cdots + Pr[\tau_r \not\equiv 1] \leq 2^s(1/2)^{s+1} = 1/2$$

Thus,

$$Pr[\tau_1 \equiv 1 \wedge \cdots \wedge \tau_r \equiv 1] \geq 1/2$$

Therefore, there is an assignment that converts the circuit $C_k^{m,2}$ to the circuit $C_{k-1}^{m,m}$ and eliminates all gates in the bottom level of the circuit C in whose input there are more than s negative literals. Let the circuit obtained from C by this assignment be C'.

Now partition the input of the function $f_{k-1}^{m,m}$ into groups of m variables each such that each group corresponds to the inputs to a bottom level gate of the tree circuit $C_{k-1}^{m,m}$. Randomly pick half of the variables in each group and assign them 1. The circuit $C_{k-1}^{m,m}$ under such an assignment is converted to the circuit $C_{k-1}^{m,m/2}$ defining the function $f_{k-1}^{m,m/2}$.

Let σ be an OR gate in the bottom level of the circuit C' with more than s positive literals in its input, then by Lemma 4.2,

$$Pr[\sigma \not\equiv 1] \leq (1/2)^{s+1}$$

Let $\sigma_1, \cdots, \sigma_t$, $t \leq 2^s$, be all the gates in the bottom level of the circuit C' with more than s positive literals in their input, then

$$Pr[\sigma_1 \not\equiv 1 \vee \cdots \vee \sigma_t \not\equiv 1] \leq Pr[\sigma_1 \not\equiv 1] + \cdots + Pr[\sigma_t \not\equiv 1] \leq 2^s(1/2)^{s+1} = 1/2$$

Thus,

$$Pr[\sigma_1 \equiv 1 \wedge \cdots \wedge \sigma_t \equiv 1] \geq 1/2$$

Therefore, there is an assignment that converts the circuit $C_{k-1}^{m,m}$ to the circuit $C_{k-1}^{m,m/2}$ and eliminates all gates in the bottom level of C' that have more than s positive literals in their input. Let the circuit obtained from C' by this assignment be C''.

Since each gate in the bottom level of the circuit C'' has neither more than s negative literals nor more than s positive literals in its input, the bottom fan-in of the circuit C'' is at most $2s = \frac{1}{6\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}$, which is smaller than $\frac{m/2}{25e(k-1)\log m}$ for sufficiently large m. Thus, we have constructed a circuit C'' of depth $k-1$, bottom fan-in less than $\frac{m/2}{25e(k-1)\log m}$, and size bounded by $2^s = 2^{\frac{1}{12\sqrt{2(k-1)}}\sqrt{\frac{m}{\log m}}}$ such that C'' computes the function $f_{k-1}^{m,m/2}$. This contradicts Theorem 3.3. □

The following corollary will be used in Section 6.

Corollary 4.4 *The function $f_k^{m,2}$ can be computed by a circuit of depth k, linear size, and bottom fan-in 2, but cannot be computed by any depth $k-1$ circuit of polynomial size.*

Corollary 4.5 *For each pair of integers $k, c > 1$, there are functions that are computable by circuits of linear size and depth k with bottom fan-in c but require exponential size for circuits of depth $k-1$.*

5 Trade-off between bottom fan-in and size

We first summarize the results in the previous two sections in the following theorem.

Theorem 5.1 *For all integers $b \geq 2$ and sufficiently large m, the function $f_k^{m,b}$ can be computed by a depth k circuit of linear size and bottom fan-in b, but requires size larger than $2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}}$ for depth k circuits of bottom fan-in $\frac{b}{25ek\log m}$.*

PROOF. For the case $2 \leq b \leq k\log m$, since $\frac{b}{25ek\log m} < 1$, the theorem is implied by Theorem 4.3. The case $k\log m < b \leq \sqrt{km\log m/2}$ is proved in Theorem 3.2. For the case $\sqrt{km\log m/2} < b < 2\sqrt{km\log m/2}$, since $\frac{b}{25ek\log m} \leq \frac{\sqrt{km\log m/2}}{12k\log m}$, the theorem is implied by Theorem 3.2. Finally, the case $b \geq 2\sqrt{km\log m/2}$ is proved by Theorem 3.3. □

A number of important consequences follow directly ¿from Theorem 5.1.

Theorem 5.2 *For any integers $k \geq 1$ and $h \geq 1$, and for any real number r, there are functions that are computable by circuits of linear size and depth k with bottom fan-in $O(\log^h n)$, but require exponential size for depth k circuits of bottom fan-in $r \log^{h-1} n$.*

PROOF. Let $b = 25ek(k-1)^{h-1} r \log^h m$. Note that in this case, the number of variables in the function $f_k^{m,b}$ is $n \leq m^{k-1}$ for m large enough. Thus, $\log m \leq \log n \leq (k-1)\log m$. By the definition, the function $f_k^{m,b}$ can be computed by a depth k and linear size circuit with bottom fan-in $b = O(\log^h n)$. On the other hand, according to Theorem 5.1, the function $f_k^{m,b}$ requires exponential size for depth k circuits whose bottom fan-in is $r(k-1)^{h-1}\log^{h-1} m$. Note that $r(k-1)^{h-1}\log^{h-1} m$ is at least as large as $r \log^{h-1} n$. □

By more careful selections of the bottom fan-in b in Theorem 5.1, combined with a padding technique, we are able to obtain general results for the trade-off between circuit size and circuit bottom fan-in. We illustrate this technique by the following theorem, which can be easily extended to other cases using the same technique.

Theorem 5.3 *For any integer $k \geq 1$ and for any real number $\epsilon > 0$, there is a function F_k^ϵ that is computable by a circuit of linear size and depth k with bottom fan-in $\log n$, but requires superpolynomial size for depth k circuits of bottom fan-in $O(\log^{1-\epsilon} n)$.*

PROOF. Choose h such that $\frac{h}{h+1} > 1 - \epsilon$, and then use Theorem 5.2 to choose a function $f_k^{m,b}$ of $\leq m^{k-1}$ variables which can be computed by a depth k circuit of linear size and bottom fan-in $b = 25ek \log^{h+1} m$ but requires size

$$2^{\frac{1}{12\sqrt{2k}}\sqrt{\frac{m}{\log m}}} \tag{1}$$

when the bottom fan-in is $\leq \log^h m$.

Now make the function $f_k^{m,b}$ formally the function F_k^ϵ of $n = 2^{25ek \log^{h+1} m}$ variables by adding dummy variables that are not used. The theorem now follows for the function F_k^ϵ since the size bound (1) is superpolynomial in n and $c \log^{1-\epsilon} n < \log^h m$ for any fixed constant c when m is sufficiently large. □

In particular, if we let $\epsilon = 1$ and $h = 1$, then we obtain the following corollary that will be used in Section 6.

Corollary 5.4 *For any integer $k \geq 1$, there is a function F_k that is computable by a circuit of linear size and depth k with bottom fan-in $\log n$, but requires superpolynomial size for depth k circuits of bottom fan-in $O(1)$.*

6 Input read-modes of Turing machines

An important application of the above investigation is to the input read-modes of a sublinear-time alternating Turing machine, which is an important computational model in the study of complexity classes.

To make sublinear-time Turing machines meaningful, we allow a Turing machine to have a *random access input tape* plus a read-write *input address tape*, such that the Turing machine has access to the bit of the input tape denoted by the contents of the input address tape.

An *$O(\log n)$-time alternating Turing machine* (log-*time ATM*) is defined as an extension of the $O(\log n)$-time deterministic Turing machines in the usual way [9]. Given an input, the computation of a log-time ATM M can be represented by an $\wedge$-$\vee$ tree. Each computation path in the $\wedge$-$\vee$ tree can be divided into *phases*, which are the maximal subpaths in which M does not make alternations. The first configuration in each phase is called an *alternation* (configuration). In particular, the starting configuration of M is always an alternation.

A number of input read-modes for sublinear-time alternating Turing machines have appeared in the literature. The standard input read-mode introduced by Chandra, Kozen, and Stockmeyer [9] allows a computation path of an $O(\log n)$-time alternating Turing machine to read up to $\Theta(\log n)$ input bits. Ruzzo [16] proposed an input read-mode in which each computation path can read at most one input bit and the reading must be performed at the end of the path. An input read-mode studied by Sipser [17] insists that the input address tape be always reset to blank after each input reading so that each input reading takes time $\Omega(\log n)$.

It can be shown that many complexity classes such as NC^k for $k \geq 1$ and AC^k for $k \geq 0$ remain the same for all these input read-modes of alternating Turing machines. On the other hand, it was unknown whether these input read-modes affect the classes of lower complexity such as the levels in the logarithmic time hierarchy. Recently, Cai and Chen [6] have demonstrated how each level of the logarithmic time hierarchy based on each of the above input read-modes can be characterized by a uniform family of circuits. Combining these characterizations with the separation results given in the previous sections, we are able to show that all these input read-modes are distinct.

Formally, the *logarithmic time hierarchy* is defined to be the union of the following classes:

$$\Pi_1, \Pi_2, \cdots, \Pi_k, \cdots$$

where Π_k is the class of languages accepted by a log-time ATM that always starts with an $\wedge$-state and makes at most k alternations.

The above definition ignores the input read-modes of the log-time ATMs and thus is not very precise. To be more precise, we will call

$$\Pi_1^U, \Pi_2^U, \cdots, \Pi_k^U, \cdots$$

the logarithmic time hierarchy *based on Chandra-Kozen-Stockmeyer's model*,

$$\Pi_1^R, \Pi_2^R, \cdots, \Pi_k^R, \cdots$$

the logarithmic time hierarchy *based on Ruzzo's model*, and

$$\Pi_1^S, \Pi_2^S, \cdots, \Pi_k^S, \cdots$$

the logarithmic time hierarchy *based on Sipser's model*, where Π_k^U (resp. Π_k^R, Π_k^S) is the class of languages accepted by a log-time ATM based on Chandra-Kozen-Stockmeyer's input read-mode (resp. on Ruzzo's input read-mode, on Sipser's input read-mode) that always starts with an $\wedge$-state and makes at most k alternations.

Theorem 6.1 ([6]) *For all integers $k \geq 1$,*

(1). *If a language L is in the class Π_k^R, then L is accepted by a Π_k^{poly}-family of circuits;*

(2). *If a language L is in the class Π_k^S, then L is accepted by a $\Pi_k^{poly,c}$-family of circuits for some constant c;*

(3). *If a language L is in the class Π_k^U, then L is accepted by a $\Pi_k^{poly,d\log n}$-family of circuits for some constant d;*

According to the definitions, it is easy to see that $\Pi_k^R \subseteq \Pi_k^S \subseteq \Pi_k^U$. A proof for the inclusion $\Pi_k^U \subseteq \Pi_{k+1}^R$ can be found in [6]. Our main result for this section is that all these inclusions are strict, as proved in the following theorem.

Theorem 6.2 *For any integer $k \geq 1$, we have*

$$\Pi_k^R \subset \Pi_k^S \subset \Pi_k^U \subset \Pi_{k+1}^R$$

where $\subset$ means "proper subset".

Proof.
(1). $\Pi_k^R \subset \Pi_k^S$.

It has been proved by Cai and Chen in [6] that $\Pi_1^R \subset \Pi_1^S$. Thus, we only need to prove the strict inclusion for $k \geq 2$.

Without loss of generality, we suppose that the output gate of the tree circuit $C_{k+1}^{m,2}$ defining the function $f_{k+1}^{m,2}$ is an AND gate (otherwise, we consider the negation of the function $f_{k+1}^{m,2}$). Moreover, to make the operations such as m^k, $\log m$, and $\sqrt{m}$ feasible within $O(\log n)$ deterministic time, we consider only the case where m is a power of 2.

Let S_1 be the language whose characteristic function is given by the functions $f_{k+1}^{m,2}$, where m is a power of 2 (in particular, a string is not in the set S_1 if its length does not match the number of variables for any such function $f_{k+1}^{m,2}$). We first construct a log-time ATM M_1 that accepts the set S_1 as follows. On input x, M_1 first computes the length n of x. This can be done in deterministic $O(\log n)$ time by reading $O(\log n)$ input bits [2]. Then M_1 verifies that $n = 2m^{k-1}\sqrt{m/\log m}$ for some integer m that is a power of 2. After this, M_1 simply traces the tree circuit $C_{k+1}^{m,2}$ defining the function $f_{k+1}^{m,2}$, except that in the kth phase, M_1 reads the two consecutive input bits for the corresponding bottom level gate and directly computes the value for the gate. Since the output gate of the circuit $C_{k+1}^{m,2}$ is an AND gate, the log-time ATM M_1 starts with an $\wedge$-state and makes at most k alternations.

According to our assumption, $k \geq 2$. Thus, the log-time ATM M_1 reads at most two input bits in its last phase. By Theorem 3.1 in Cai and Chen [6], M_1 can be simulated by a log-time ATM based on Sipser's input read-mode that always starts with an $\wedge$-state and makes at most k alternations. This proves that the set S_1 is in the class Π_k^S.

Suppose that S_1 is also in the class Π_k^R. Then by Theorem 6.1(1), S_1 is accepted by a Π_k^{poly}-family of circuits. Thus, for any integer m that is a power of 2, the function $f_{k+1}^{m,2}$ is computable by a depth k circuit of polynomial size. This contradicts Corollary 4.4.

This shows $\Pi_k^R \subset \Pi_k^S$.

(2). $\Pi_k^S \subset \Pi_k^U$.

The proof is similar to that for Case (1). Consider the function F_{k+1} in Corollary 5.4, which is a function of n variables obtained from the function $f_{k+1}^{m,b}$, $b = 25e(k+1)\log^2 m$ by adding dummy variables, where $n = 2^{25e(k+1)\log^2 m}$. We also make the similar assumptions as we did for Case (1). Thus, the output gate of the tree circuit $C_{k+1}^{m,b}$ defining the function $f_{k+1}^{m,b}$ is an AND gate, and m is a power of 2. Under these assumptions, it is easy to see that the set S_2 whose characteristic function is given by the functions F_{k+1} in Corollary 5.4 is in the class Π_k^U: A log-time ATM M_2 first verifies the length of the input and traces the tree circuit $C_{k+1}^{m,b}$ defining the corresponding function $f_{k+1}^{m,b}$ except that in the kth phase, M_2 reads directly a consecutive block of b input bits that are the inputs to the corresponding bottom level gate. Note that the b consecutive input bits can be read in deterministic $O(b + \log n)$ time [4], and that b is logarithmic in the input length n of the function F_{k+1}. This proves that the language S_2 is in the class Π_k^U.

Suppose that S_2 is also in the class Π_k^S. By Theorem 6.1(2), the language S_2 is accepted by a $\Pi_k^{poly,c}$-family of circuits for some constant c. That is, the function F_{k+1} is computable by a depth $k+1$ and bottom fan-in c circuit whose size is polynomial. But this contradicts Corollary 5.4. Thus, $\Pi_k^S \subset \Pi_k^U$.

(3). $\Pi_k^U \subset \Pi_{k+1}^R$.

The proof is similar to those for the other two cases. Let S_3 be the language whose characteristic function is given by the Sipser functions f_{k+1}^m. Then the set S_3 can be accepted by a log-time ATM M_3 that always starts with an $\wedge$-state and makes at most $k+1$ alternations. Moreover, the last phase of M_3 reads at most one input bit. By Theorem 3.1 in Cai and Chen [6], M_3 can be simulated by a log-time ATM based on Ruzzo's input read mode that always starts with an $\wedge$-state and makes at most $k+1$ alternations. Thus, the set S_3 is in the class Π_{k+1}^R. On the other hand, by Theorem 6.1(3), $S_3 \in \Pi_k^U$ would imply that S_3 is accepted by a $\Pi_k^{poly,O(\log n)}$-family of circuits. That would in turn imply that the Sipser function f_{k+1}^m is computable by a depth $k+1$ circuit of polynomial size whose bottom fan-in is $O(\log n)$, contradicting Theorem 2.1.

This completes the proof. □

Corollary 6.3 *For each $k \geq 1$, the kth levels of the logarithmic time hierarchy based on Chandra-Kozen-Stockmeyer's input read-mode, Sipser's input read-mode, and Ruzzo's input read-mode are all distinct.*

Acknowledgements.

The second author would like to thank Mike Sipser for an early discussion that initializes this line of research. He is also grateful to Ken Regan for his comments and constructive discussions. Finally, the authors are especially thankful to two anonymous referees for comments and suggestions that have improved the presentation. In particular, one of the referees pointed out a technical bug in an earlier version of the present paper.

References

[1] M. Ajtai, Σ_1^1-formulae on finite structures, *Ann. Pure Appl. Logic 24*, (1983), pp. 1-48.

[2] D. Barrington, N. Immerman, and H. Straubing, On uniformity within NC^1, *J. Comput. System Sci. 41*, (1990), pp. 274-306.

[3] R. B. Boppana and M. Sipser, The complexity of finite functions, in J. van Leeuwen, ed., *Handbook of Theoretical Computer Science Vol. A*, Elsevier, Amsterdam, 1990, pp. 757-804.

[4] S. R. Buss, The Boolean formula value problem is in ALOGTIME, *Proc. 19th Annual ACM Symposium on Theory of Computing*, (1987), pp. 123-131.

[5] L. Cai and J. Chen, Fixed parameter tractability and approximability of *NP*-hard optimization problems, *Proc. 2rd Israel Symposium on Theory of Computing and Systems*, (1993), pp. 118-126. Journal version submitted.

[6] L. Cai and J. Chen, On input read-modes of alternating Turing machines, *Theoretical Computer Science 148*, (1995), pp. 33-55.

[7] L. Cai and J. Chen, On the amount of nondeterminism and the power of verifying, *SIAM Journal on Computing*, to appear.

[8] L. Cai, J. Chen, R. G. Downey, and M. R. Fellows, On the structure of parameterized problems in NP, *Information and Computation 123*, (1995), pp. 38-49.

[9] A. K. Chandra, D. C. Kozen, and L. J. Stockmeyer, Alternation, *J. Assoc. Comput. Mach. 28*, (1981), pp. 114-133.

[10] J. Chen, Characterizing parallel hierarchies by reducibilities, *Information Processing Letters 39*, (1991), pp. 303-307.

[11] S. Cook, A taxonomy of problems with fast parallel algorithms, *Information and Control 64*, (1985), pp. 2-22.

[12] M. Furst, B. Saxe, and M. Sipser, Parity, circuits, and the polynomial-time hierarchy, *Math. Systems Theory 17*, (1984), pp. 13-27.

[13] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics: A Foundation for Computer Science*, Addison-Wesley, Reading, MA, 1989.

[14] J. Håstad, *Computational limitations for small-depth circuits*, The MIT Press, Cambridge, MA, 1986.

[15] J. Håstad, Almost optimal lower bounds for small depth circuits, in S. Micali, ed., *Advances in Computing Research 5*, JAI Press Inc., Greenwich, CT, 1989, pp. 143-170.

[16] W. L. Ruzzo, On uniform circuit complexity, *J. Comput. System Sci. 22*, (1981), pp. 365-383.

[17] M. Sipser, Borel sets and circuit complexity, *Proc. 15th Annual ACM Symposium on Theory of Computing*, (1983), pp. 61-69.

[18] A. C. Yao, Separating the polynomial-time hierarchy by oracles, *Proc. 26th Annual IEEE Symposium on Foundations of Computer Science*, (1985), pp. 1-10.