Proceedings of the 2014 ACM Symposium on Theory of Computing

Formulas vs. Circuits for Small Distance Connectivity

[Extended Abstract]*

Benjamin Rossman†
National Institute of Informatics
2-1-2 Hitotsubashi, Chiyoda-ku
Tokyo 101-8430, Japan
rossman@nii.ac.jp

ABSTRACT

We give the first super-polynomial separation in the power of bounded-depth boolean formulas vs. circuits. Specifically, we consider the problem Distance $k(n)$ Connectivity, which asks whether two specified nodes in a graph of size n are connected by a path of length at most $k(n)$. This problem is solvable (by the recursive doubling technique) on **circuits** of depth $O(\log k)$ and size $O(kn^3)$. In contrast, we show that solving this problem on **formulas** of depth $\log n/(\log\log n)^{O(1)}$ requires size $n^{\Omega(\log k)}$ for all $k(n) \leq \log\log n$. As corollaries:

(i) It follows that polynomial-size circuits for Distance $k(n)$ Connectivity require depth $\Omega(\log k)$ for all $k(n) \leq \log\log n$. This matches the upper bound from recursive doubling and improves a previous $\Omega(\log\log k)$ lower bound of Beame, Impagliazzo and Pitassi [BIP98].

(ii) We get a tight lower bound of $s^{\Omega(d)}$ on the size required to simulate size-s depth-d circuits by depth-d formulas for all $s(n) = n^{O(1)}$ and $d(n) \leq \log\log\log n$. No lower bound better than $s^{\Omega(1)}$ was previously known for any $d(n) \nleq O(1)$.

Our proof technique is centered on a new notion of *pathset complexity*, which roughly speaking measures the minimum cost of constructing a set of (partial) paths in a universe of size n via the operations of union and relational join, subject to certain density constraints. Half of our proof shows that bounded-depth formulas solving Distance $k(n)$ Connectivity imply upper bounds on pathset complexity. The other half is a combinatorial lower bound on pathset complexity.

*A full version of this paper is available at `http://eccc.hpi-web.de/report/2013/169/`

†Supported by JST ERATO Kawarabayashi Large Graph Project.

STOC '14, May 31 - June 03 2014, New York, NY, USA

http://dx.doi.org/10.1145/2591796.2591828.

Categories and Subject Descriptors

F.1.3 [**Computation by Abstract Devices**]: Complexity Measures and Classes

General Terms

Theory

1. INTRODUCTION

Understanding the relative power of formulas versus circuits is a central challenge in complexity theory, especially in the important boolean setting. Whereas boolean circuits are the most general non-uniform model of computation, there is a strong intuition that boolean formulas (= tree-like circuits with fan-out 1) are a very weak model of computation. Many natural problems solvable by small circuits, such as st-connectivity, are believed to require large formulas. However, no super-polynomial gap between the formula complexity and circuit complexity of any problem has ever been established.

Question 1. *Are polynomial-size boolean circuits strictly more powerful than polynomial-size boolean formulas?*

There are two versions of Question 1 in the **uniform** and **non-uniform** settings.[1] In terms of complexity classes, this is equivalent to asking whether uniform-NC^1 (resp. NC^1) is a proper subclass of P (resp. $\mathsf{P/poly}$).[2] As we discuss next, both the uniform and non-uniform versions of this question are wide open.

An obvious prerequisite of separating uniform-NC^1 from P is a super-polynomial lower bound on the formula complexity of *any* explicit boolean function. However, despite the fact that *almost all* boolean functions have formula complexity $\Omega(2^n/\log n)$ by a classic theorem of Riordan and Shannon [18], the best lower bound for any explicit function, due to Håstad [7], is only $\Omega(n^{3-o(1)})$. Unfortunately, n^3 is known

[1]Whenever we speak of a circuit (or formula), this is understood to mean a sequence $(C_n)_{n=1}^{\infty}$ of circuits, one for each input size n. In the *uniform* setting, there is an underlying algorithm which, given 1^n as input, outputs a description of the circuit C_n. In the *non-uniform* setting, C_n are arbitrary. All bounds mentioned in this paper may be interpreted in the stronger sense: uniform upper bounds and non-uniform lower bounds.

[2]By Spira's Theorem [25], NC^1 is equivalent to the class of languages recognized by polynomial-size boolean formulas (of unbounded depth).

to be the limit of existing techniques, and it appears that any improvement will require a major breakthrough.

In the non-uniform setting, the situation is no better. By a striking theorem of Savický and Woods [21], for every constant $k > 1$, almost all boolean functions with formula complexity $\leq n^k$ have circuit complexity $\geq n^k/k$. This shows that NC^1 cannot be separated from $\mathsf{P/poly}$ by a straightforward counting argument (in contrast with results like the Circuit Size Hierarchy Theorem, see [10]). Other than by counting arguments, it is not clear how to take advantage of non-uniformity.

In short, it appears that we are a long way from answering Question 1. In the meantime, we can hope to gain insight by studying the question of formulas vs. circuits in *restricted settings* where strong lower bounds are available. In particular, Question 1 has natural analogues in both the **monotone** setting and the **bounded-depth** boolean setting, where exponential lower bounds have been around for decades. However, as we will explain, while question of monotone formulas vs. circuits has been settled for 25 years, essentially nothing was known in bounded-depth setting prior to the results of this paper.

Monotone Formulas vs. Circuits

Recall that *monotone* circuits are boolean circuits without negation gates. The separation of monotone formulas from monotone circuits was shown by Karchmer and Wigderson [11] via a lower bound for directed st-connectivity (*STCONN*).

Theorem 1. *Monotone formulas solving STCONN require size $n^{\Omega(\log n)}$.*

As it was already known that *STCONN* has polynomial-size monotone circuits, Theorem 1 implies the separation of monotone classes mNC^1 and mP (in fact, it shows $\mathsf{mNC}^1 \neq \mathsf{mAC}^1$). (In a notable recent development, Potechin [13] showed that monotone switching networks for *STCONN* require size $n^{\Omega(\log n)}$. This result strengthens Theorem 1 and implies the sharper separation $\mathsf{mL} \neq \mathsf{mNL}$.)

Bounded-Depth Formulas vs. Circuits

The *bounded-depth* setting refers to the class of unbounded fan-in boolean circuits and formulas of depth $\leq d(n)$ for some (not necessarily constant) function $d : \mathbb{N} \to \mathbb{N}$. Unlike the monotone setting, the question of bounded-depth formulas vs. questions gives a natural approach to Question 1: by comparing the power of depth-d formulas vs. depth-d circuits, we can hope to get a separation for as large a depth $d(n)$ as possible, noting that a super-polynomial separation for any $d(n) = \log n$ would imply $\mathsf{NC}^1 \neq \mathsf{AC}^1$ (answering Question 1).

We write $\mathsf{Circuit}(s, d)$ (resp. $\mathsf{Formula}(s, d)$) for the class of languages computable by unbounded fan-in boolean circuits (resp. formulas) of size $\leq s(n)$ and depth $\leq d(n)$. Consider the elementary fact that $\mathsf{Circuit}(s, d) \subseteq \mathsf{Formula}(s^d, d)$, that is, every depth-$d$ circuit of size s is equivalent to a depth-d formula of size $\leq s^d$. In the naive simulation of circuits by formulas, we simply replace overlapping subcircuits with non-overlapping copies until the circuit becomes a tree. Note that this give a slightly better upper bound of $(\text{fan-in})^d$. It is natural to ask: is this naive simulation of depth-d circuits by depth-d formulas asymptotically optimal? To make this question meaningful, we focus on the case where $s(n)$ is any $n^{O(1)}$ and $d(n) \leq \log n$. Thus, $\mathsf{Circuit}(n^{O(1)}, d) \subseteq \mathsf{Formula}(n^{O(d)}, d)$ and we can ask whether $n^{O(d)}$ can be improved to $n^{o(d)}$.

Question 2. *For which functions $d(n) \leq \log n$ do we have*

$$\mathsf{Circuit}(n^{O(1)}, d) \nsubseteq \mathsf{Formula}(n^{o(d)}, d)? \qquad (*)$$

On the basis of problems like *STCONN*, we conjecture that $(*)$ holds for all $d(n) \leq \log n$. Of course, since this (more than) implies $\mathsf{NC}^1 \neq \mathsf{AC}^1$, we should not expect to prove $(*)$ all the way to depth $\log n$ anytime soon. On the other hand, more modest depths like $O(\log \log n)$ are well within the range of techniques like switching lemmas (after all, the super-polynomial lower bounds for parity extend to depth $o(\log n / \log \log n)$ [6]). For this reason, it might seem that $(*)$ is the kind of statement that ought to be known (or follow from known results) for modest but super-constant $d(n)$. (Note that $(*)$ is trivial for constant $d(n) = O(1)$, as $n^{o(d)} = n^{o(1)}$.) However, it turns out that the status of $(*)$ was entirely unknown for all $d(n) \nleq O(1)$. Even the weakest possible separation $\mathsf{Circuit}(n^{O(1)}, d) \nsubseteq \mathsf{Formula}(n^{O(1)}, d)$ (i.e. $\mathsf{Formula}(n^{O(1)}, d) \subsetneq \mathsf{Circuit}(n^{O(1)}, d)$) was not known to hold for any $d(n) \nleq O(1)$. In this paper, we improve this state of affairs by showing that $(*)$ holds for all $d(n) \leq \log \log \log n$ (Corollary 2).

At this point, we should address the question: why have the previous techniques (in particular, switching lemmas [6] and approximation by low-degree polynomials [15, 24]) failed to distinguish formulas from circuits? In other words, why don't these techniques imply stronger lower bounds for depth-d formulas, as compared to depth-d circuits? One explanation is the style of bottom-up depth-reduction arguments that the previous technique employ. A second, complementary explanation is the lack of successful top-down lower bounds—in particular, via Karchmer-Wigderson games [11]—in the non-monotone boolean setting (with the exception of the depth-3 lower bound of Jukna, Pudlák and Håstad [8]). Our technique gets around the limitations of previous techniques by a novel combination of bottom-up and top-down arguments.

Distance $k(n)$ Connectivity

As with the separation of monotone formulas vs. circuits in [11], our separation of bounded-depth formulas vs. circuits comes by way of a lower bound for (a parameterized version of) st-connectivity. As Wigderson wrote in his excellent survey on graph connectivity [27], "Of all computational problems, graph connectivity is the one that has been studied on the largest variety of computational models, such as Turing machines, PRAMs, Boolean circuits, decision trees and communication complexity. It has proven a fertile test case for comparing basic resources such as time vs. space, nondeterminism vs. randomness vs. determinism, and sequential vs. parallel computation." There has been some significant progress in the 20 years since [27]. One notable result is Reingold's theorem [17] that *USTCONN* (undirected st-connectivity) $\in \mathsf{DSPACE}(\log n)$. However, many questions remain open. Chief among these is the space complexity of *STCONN*. Savitch's theorem [22] that *STCONN* $\in \mathsf{DSPACE}(\log^2 n)$ is still the best known upper bound.

As for lower bounds for *STCONN*, in addition to various results in monotone models of computation [11, 13, 14, 23, 26], there are results on structured models of computations whose basic operations manipulate pebblings on graphs. One result of this type, due to Edmonds, Poon and Achlioptas [4], gives a tight space lower bound of $\Omega(\log^2 n)$ on the NNJAG model. Another interesting result, in the unusual restricted model of arithmetic circuits with $\times$ gates of odd fan-in, is a tight lower bound of $n^{\Omega(\log n)}$ for *STCONN* (or more accurately its algebraic cousin, iterated matrix multiplication) was shown by Nisan and Wigderson [12] using the method of partial derivatives.

In this paper, we consider a version of *STCONN* parameterized by distance. For a function $k : \mathbb{N} \to \mathbb{N}$ with $k(n) \le n$, *distance $k(n)$ connectivity*, denoted $STCONN(k(n))$, is the following problem: given a directed graph with n vertices and specified vertices s and t, determine whether or not there is a path of length at most $k(n)$ from s to t. (The directed and undirected versions of distance $k(n)$ connectivity are essentially equivalent.) The recursive doubling (a.k.a. repeated squared) method of Savitch [22] shows that $STCONN(k(n))$ has (semi-unbounded, monotone) circuits of size $O(kn^3)$ and depth $2\log k$. (At the expense of larger depth, one can get smaller circuits of size $O(kn^{2.38})$ using fast matrix multiplication.)

Every algorithm for $STCONN(k(n))$ "scales up" to an algorithm for *STCONN* by recursive kth powering.[3] Note the implication:

$$STCONN(k(n)) \in \mathsf{Circuit}(s,d) \implies STCONN \in \mathsf{Circuit}(n^{O(1)} \cdot s, \tfrac{\log n}{\log k} \cdot d).$$

It follows that if $STCONN(k(n))$ has polynomial-size circuits of depth $o(\log k)$, then *STCONN* has polynomial-size circuits of depth $o(\log n)$ and hence $STCONN \in \mathsf{DSPACE}(o(\log^2 n))$. This observation strongly motivates the following

Question 3. *What is the minimum depth of polynomial-size circuits solving $STCONN(k(n))$?*

Furst, Saxe and Sipser [5] showed that $STCONN \notin \mathsf{AC}^0$ via the reduction from parity to *STCONN*. Via the same reduction, it follows from the parity lower bound of Håstad [6] that $STCONN(k(n)) \notin \mathsf{AC}^0$ for all $k(n) \not\le \log^{O(1)} n$. However, this says nothing when $k(n) \le \log^{O(1)} n$.

Ajtai [1] proved the first lower bound for small distances $k(n)$, showing that $STCONN(k(n)) \notin \mathsf{AC}^0$ for all $k(n) \not\le O(1)$. Via an explicit vetsion of Ajtai's non-constructive proof, Bellantoni, Pitassi and Urquhart [3] proved a lower bound of $\Omega(\log^* k)$ on the depth of polynomial-size circuits solving $STCONN(k(n))$. This was subsequently improved to $\Omega(\log\log k)$ for all $k(n) \le \log^{O(1)} n$ by Beame, Impagliazzo and Pitassi [2], using a special-purpose "connectivity switching lemma" tailored to $STCONN(k(n))$. It was left as an open problem to further narrow the gap between the $O(\log k)$ and $\Omega(\log\log k)$ upper and lower bounds. In this paper, we completely close this gap by proving a lower bound of $\Omega(\log k)$ for all $k(n) \le \log\log n$ (Corollary 1). (While our current proof is restricted to $k(n) \le \log\log n$, we believe this can be extended $k(n) \le \log^{O(1)} \log n$ as in [2].) The significance of this result is that, for small but super-constant $k(n)$, we rule out the possibility of showing that $STCONN \in \mathsf{DSPACE}(o(\log^2 n))$ by constructing polynomial-size circuits for $STCONN(k(n))$ of depth $o(\log k)$.

[3] Conversely, every lower bound for *STCONN* "scales down" to a lower bound for $STCONN(k(n))$. In particular, Theorem 1 implies that monotone formulas solving $STCONN(k(n))$ require size $n^{\Omega(\log k)}$.

2. OUR RESULTS

Our main theorem is a tight lower bound on the size of bounded-depth formulas solving distance $k(n)$ connectivity.

Theorem 2. *Formulas of depth $\frac{\log n}{(\log\log n)^{O(1)}}$ solving $STCONN(k(n))$ require size $n^{\Omega(\log k)}$ for all $k(n) \le \log\log n$.*

We restate Theorem 2 with more specific parameters in §4. We remark that the lower bound of Theorem 2 applies to formulas solving $STCONN(k(n))$ in the most natural average-case sense (see §7).

The following two corollaries of Theorem 2 were already mentioned in the introduction. As discussed, these corollaries answer Questions 2 and 3 for a limited range of $d(n)$ and $k(n)$.

Corollary 1. *Polynomial-size circuits solving $STCONN(k(n))$ require depth $\Omega(\log k)$ for all $k(n) \le \log\log n$.*

Proof. For contradiction, assume C is a circuit of size $s(n) = n^{O(1)}$ and depth $d(n) = o(\log k)$ solving $STCONN(k(n))$ for some $k(n) \le \log\log n$. By the naive simulation of circuits by formulas, C is equivalent to a depth-d formula of size $\le s^d = n^{o(\log k)}$. But since $d(n) = o(\log\log\log n) \ll \frac{\log n}{(\log\log n)^{O(1)}}$, we get a contradiction with Theorem 2. $\square$

Corollary 2. *It is impossible to simulate polynomial-size depth-d circuits by depth-d formulas of size $n^{o(d)}$ (that is, the optimal separation $\mathsf{Circuit}(n^{O(1)}, d) \nsubseteq \mathsf{Formula}(n^{o(d)}, d)$ holds) for all $s(n) = n^{O(1)}$ and $d(n) \le \log\log\log n$.*

Proof. The separating language is $STCONN(k(n))$ where $k(n) = 2^{d(n)/2}$ ($\le \log\log n$). The upper bound $STCONN(k(n)) \in \mathsf{Circuit}(n^{O(1)}, d)$ comes from the circuits (of depth $2\log k$) which implement recursive doubling. The lower bound $STCONN(k(n)) \notin \mathsf{Formula}(n^{o(d)}, d)$ is by Theorem 2, noting that $d(n) \le \log\log\log n \ll \frac{\log n}{(\log\log n)^{O(1)}}$. $\square$

3. PROOF OVERVIEW

Our proof technique is centered on a new notion of *pathset complexity*. Informally, a *pathset* is a subset $\mathcal{A} \subseteq [n]^{k+1}$ whose elements represent potential paths of length k in a graph of size n. The *pathset complexity* of $\mathcal{A}$, denoted $\chi(\mathcal{A})$, measures of the minimum number of operations required to construct $\mathcal{A}$ via unions ($\cup$) and relational join ($\bowtie$), subject to certain density constraints. (The formal definition of $\chi(\mathcal{A})$, given in §5, is not important for this overview.)

The proof of Theorem 2 has two parts. Part 1 shows that every bounded-depth formula F solving $STCONN(k(n))$ implies an upper bound on the pathset complexity of a certain (random) pathset $\mathcal{A}^\Gamma$. Part 2 is a general lower bound on $\chi(\mathcal{A})$ for arbitrary pathsets $\mathcal{A}$. Combining these two parts, we get the desired $n^{\Omega(\log k)}$ lower bound on the size of F.

Before explaining Parts 1 and 2 in more detail, we state the key property of $STCONN(k(n))$ which our proof exploits. Instances for $STCONN(k(n))$ are directed graphs with vertex set $[n]$ and distinguished vertices s and t (without loss of generality $s = 1$ and $t = 2$). An *st-path* is a sequence $(x_0, \dots, x_k) \in [n]^{k+1}$ such that $x_0 = s$ and $x_k = t$ and $x_i \neq x_j$ for all $i \neq j$.

Denote by Γ the random directed graph with edge probability $1/n$. Note that $1/n$ is a sub-critical edge density for *STCONN*$(k(n))$, that is, almost surely Γ contains no *st*-path of length k. Define $\mathcal{A}^\Gamma$ as the set of *st*-paths $(x_0, \dots, x_k) \in [n]^{k+1}$ such that

- $(x_0, x_1), \dots, (x_{k-1}, x_k)$ are non-edges of Γ,
- $\Gamma \cup \{(x_0, x_1), \dots, (x_{k-1}, x_k)\}$ contains a unique *st*-path of length k (namely, $(x_0, \dots, x_k)$).

The average-case property of *STCONN*$(k(n))$ that our proof exploits is:

Average-Case Property of *STCONN*$(k(n))$ (§6.3).
Almost surely, $\mathcal{A}^\Gamma$ contains 99% of st-paths of length k.

We now state Parts 1 and 2 of the proof of Theorem 2 in more detail.

Part 1: Reduction to Pathset Complexity (§6).
Suppose F is a formula of depth $\log n/(\log\log n)^{O(1)}$ solving STCONN$(k(n))$. Then almost surely (w.r.t. Γ)

$$\text{size}(F) \geq 2^{-O(k^2)} n^{-O(1)} \chi(\mathcal{A}^\Gamma). \tag{1}$$

Part 2: Pathset Complexity Lower Bound (see full paper).
For all pathsets $\mathcal{A} \subseteq [n]^{k+1}$, writing $\delta(\mathcal{A}) := |\mathcal{A}|/n^{k+1}$ for the density of $\mathcal{A}$,

$$\chi(\mathcal{A}) \geq 2^{-O(2^k)} n^{\Omega(\log k)} \delta(\mathcal{A}). \tag{2}$$

Combining (1) and (2) with $\delta(\mathcal{A}^\Gamma) \geq .99n^{-2}$ (by the average-case property), we get the lower bound

$$\text{size}(F) \geq 2^{-O(2^k)} n^{\Omega(\log k)}.$$

Since $2^{-O(2^k)} \leq n^{-O(1)}$ for $k(n) \leq \log\log n$, Theorem 2 is proved.

Part 1 builds on the technique of [19, 20]. An essential new ingredient, which distinguishes formulas from circuits, is a top-down argument (Lemma 4) relating formula size to pathset complexity.

For Part 2, we develop a combinatorial framework for studying pathset complexity. This involves analyzing the *pattern* of joins which predominates the construction of a given pathset $\mathcal{A}$. We define an auxiliary notion of *pathset complexity with respect to a pattern*, denoted $\bar{\chi}(\mathcal{A})$. Part 2 then consists of 2a and 2b:

Part 2a: From χ to $\bar{\chi}$.
For every pathset $\mathcal{A}$, there exists $\mathcal{A}' \subseteq \mathcal{A}$ such that $\chi(\mathcal{A}) \geq \bar{\chi}(\mathcal{A}')$ and $\delta(\mathcal{A}') \geq 2^{-O(2^k)}\delta(\mathcal{A})$.

Part 2b: Lower Bound for $\bar{\chi}$.
$\bar{\chi}(\mathcal{A}) \geq n^{\Omega(\log k)}\delta(\mathcal{A})$ for all pathsets $\mathcal{A}$.

Part 2a is straightforward. This move from χ to $\bar{\chi}$ is precisely where we lose the factor of $2^{O(2^k)}$, which is the reason that our main theorem is limited to $k(n) \leq \log\log n$. (If this factor can be removed, then Theorem 2 and Corollary 1 would hold up to $k(n) \leq \log^{1/3} n$ and Corollary 2 would hold up to $d(n) \leq \log\log n$.)

Part 2b is the true combinatorial lower bound at the heart the paper. The proof involves an intricate induction on patterns. Unfortunately, there is not room to include any of Part 2 in this extended abstract; see the full paper for details.

Organization of the Paper

Section 4 sets out the basic terminology and notation for the paper. Section 5 introduces the key notion of *pathset complexity*. Section 6 gives Part 1 of the proof of Theorem 2, modulo the main technical lemma which is proved in §7 of the full paper. Part 2 of the proof is given in §8–9 of the full paper. We state some conclusions and discuss future directions in Section 7. The full paper contains three appendices with supplementary material including key examples and relatively easier special cases of our main lower bound.

4. PRELIMINARIES

Let n be an arbitrary positive integer (which we view as growing to infinity). Let $[n] := \{1, \dots, n\}$. We note that, for all purposes in this paper, $[n]$ may be regarded as an arbitrary fixed set of size n. Let $k = k(n)$ and $d = d(n)$ be arbitrary functions of n. As parameters, k represents *distance* and d represents *depth*. No bound on k or d is assumed throughout the paper; assumptions like $k(n) \leq \log\log n$ are explicitly stated where needed. All constants in asymptotic notation ($O(\cdot)$, etc.) are universal (with no dependence on n, k, d).

Circuits and Formulas

The *circuits* and *formulas* considered in this paper are unbounded fan-in boolean circuits and formulas with a single output node and NOT gates at the bottom level. Formally, a *circuit* is a finite acyclic directed graph with a unique output (node of out-degree 0) where each input (node of in-degree 0) is labeled by a literal (i.e. X_i or $\overline{X}_i$) and each gate (node of in-degree ≥ 1) is labeled by AND or OR. A *formula* is a tree-like circuit in which every node other than the output has out-degree 1. The *size* of a circuit is the number of gates, while the *size* of a formula is the number of leaves. (For a formula F, the circuit-size of F equals the formula-size of F minus 1.)

Graphs

All *graphs* in this paper are directed graph $G = (V_G, E_G)$ where V_G is a (possibly empty) set and $E_G \subseteq V_G \times V_G$. The edge from v to w is written simply as vw to cut down on unnecessary parentheses.

Two important graphs in this paper are P_k (the directed path of length k) and $P_{k,n}$ (the "complete k-layered graph" with $k+1$ layers of n vertices and kn^2 edges). Formally, let $P_k = (V_k, E_k)$ where $V_k = \{v_0, \dots, v_k\}$ and $E_k = \{v_i v_{i+1} : 0 \leq i < k\}$ where $v_0, \dots, v_k$ are fixed abstract vertices. We will usually omit subscripts writing simply v and vw for arbitrary elements of V_k and E_k. To define $P_{k,n}$, we create $(k+1)n$ fresh vertices denoted v^i for each $v \in V_k$ and $i \in [n]$.

Then $P_{k,n} = (V_{k,n}, E_{k,n})$ where $V_{k,n} = \{v^i : v \in V_k,\ i \in [n]\}$ and $E_{k,n} = \{v^i w^j : vw \in E_k,\ i,j \in [n]\}$.

We refer to subgraphs $\Gamma \subseteq P_{k,n}$ with $V_\Gamma = V_{k,n}$ as *k-layered graphs.* Throughout the paper, Γ consistently represents a (random) k-layered graph, while G, H, K are reserved for subgraphs of P_k. We sometimes view Γ as the input to a circuit or formula; in this case, we identify the set of layered graphs with $\{0,1\}^N$ where N is a set of kn^2 variables indexed by elements of $E_{k,n}$.

Layered Distance $k(n)$ Connectivity

As with previous lower bounds for distance $k(n)$ connectivity [1, 2], we consider a variant of the problem on k-layered graphs. Let s, t denote vertices v_0^1, v_k^1 respectively. *Layered distance $k(n)$ connectivity* is the problem of determining whether a layered graph $\Gamma \in \{0,1\}^N$ contains a path from s to t. Following [2], we denote this problem by *DISTCONN*(k,n). There are very efficient AC^0 reductions between layered and unlayered versions of distance $k(n)$ connectivity. This allows us to restate Theorem 2 as a lower bound on *DISTCONN*(k,n):

THEOREM 2. (restated) *Solving DISTCONN(k,n) on formulas of depth $\frac{\log n}{k^3 \log\log n}$ requires size $n^{(1/6)\log k - O(1)}$ for all $k(n) \le \log\log n$.*

Remark 1. Our proof shows that Theorem 2 holds up to depth $O(\frac{\log n}{\max\{k^3 \log^2 k,\ k \log k \log\log n\}})$, which is $O(\frac{\log n}{k \log k \log\log n})$ for $k(n) \le \log^{1/3} \log n$. We state Theorem 2 with depth $\frac{\log n}{k^3 \log\log n}$ for the sake of simplicity.

Boolean Functions and Restrictions

Let $f : \{0,1\}^I \to \{0,1\}$ be a boolean function where I is an arbitrary finite set (of "variables"). We say that a variable $i \in I$ is *live* with respect to f if there exists $x \in \{0,1\}^N$ such that $f(x) \neq f(x')$ where x' equals x with its ith coordinate flipped. Let $Live(f) := \{i \in I : i \text{ is live w.r.t. } f\}$.

A *restriction* on I is any function $\theta : I \to \{0,1,*\}$. We denote by $f\lceil\theta : \{0,1\}^{\theta^{-1}(*)} \to \{0,1\}$ the function (over the "unrestricted" variables i such that $\theta(i) = *$) obtained from f by applying the restriction θ.

For $p \in [0,1]$, we write $x \in \{0,1\}^I_p$ for the random tuple $x \in \{0,1\}^I$ where $\mathbb{P}[\, x_i = 1\,] = p$ independently for all $i \in I$ (in particular, we will consider the random layered graph $\Gamma \in \{0,1\}^N_{1/n}$).

Relational Calculus

The following notation pertains to "V-ary" tuples $x \in [n]^V$ and relations $\mathcal{A} \subseteq [n]^V$ where V is an arbitrary finite set.

Definition 1. (V-TUPLES) For $x \in [n]^V$ and $S \subseteq V$, we denote by $x_S \in [n]^S$ the restriction of x to coordinates in S. For $x \in [n]^V$ and $y \in [n]^W$ where $V \cap W = \emptyset$, let $xy \in [n]^{V \cup W}$ denote the unique $z \in [n]^{V \cup W}$ such that $z_i = x_i$ for all $i \in V$ and $z_j = y_j$ for all $j \in W$; here $xy = yx$, as there is no intrinsic linear order on $V \cup W$. We adopt the convention $[n]^\emptyset = \{()\}$ where $()$ denotes the unique $\emptyset$-tuple.

Definition 2. (JOIN) For finite sets V and W and $\mathcal{A} \subseteq [n]^V$ and $\mathcal{B} \subseteq [n]^W$, the *join* of $\mathcal{A}$ and $\mathcal{B}$ is the set

$$\mathcal{A} \bowtie \mathcal{B} := \{x \in [n]^{V \cup W} : x_V \in \mathcal{A} \text{ and } x_W \in \mathcal{B}\}.$$

The join operation $\bowtie$ is a hybrid of intersection $\cap$ and cartesian product $\times$: if $V = W$ then $\mathcal{A} \bowtie \mathcal{B} = \mathcal{A} \cap \mathcal{B}$, and if $V \cap W = \emptyset$ then $\mathcal{A} \bowtie \mathcal{B}$ is the product $\mathcal{A} \times \mathcal{B}$. Note that $\mathcal{A} \bowtie \emptyset = \emptyset$ and $\mathcal{A} \bowtie \{()\} = \mathcal{A}$.

Definition 3. (DENSITY, PROJECTION, RESTRICTION) Let $\mathcal{A} \subseteq [n]^V$.

- The *density* of $\mathcal{A}$ is defined by $\delta(\mathcal{A}) := |\mathcal{A}| \,/\, n^{|V|}$.
- For $S \subseteq V$, the *S-projection* and *S-projection density* of $\mathcal{A}$ are defined by

  $$\mathrm{proj}_S(\mathcal{A}) := \{x_S : x \in \mathcal{A}\}, \quad \pi_S(\mathcal{A}) := \delta(\mathrm{proj}_S(\mathcal{A})).$$

  That is, $\pi_S(\mathcal{A}) = |\mathrm{proj}_S(\mathcal{A})| \,/\, n^{|S|}$, as δ here refers to the density of the S-ary relation $\mathrm{proj}_S(\mathcal{A}) \subseteq [n]^S$.
- For $S \subseteq V$ and $z \in [n]^{V \setminus S}$, the *$S$-restriction* of $\mathcal{A}$ at z and *maximum S-restriction density* of $\mathcal{A}$ are defined by

  $$\mathcal{A}|^z_S := \{y \in [n]^S : yz \in \mathcal{A}\}, \quad \mu_S(\mathcal{A}) := \max_{z \in [n]^{V \setminus S}} \delta(\mathcal{A}|^z_S).$$

We conclude this section with a lemma which gives some basic inequalities relating the densities of projections, restrictions and joins.

LEMMA 1. *For all $\mathcal{A} \subseteq [n]^V$ and $\mathcal{B} \subseteq [n]^W$ and $S^- \subseteq S \subseteq S^+ \subseteq V$ and $T \subseteq W$,*

(a) $\mu_{S^+}(\mathcal{A}) \le \mu_S(\mathcal{A}) \le \mu_S(\mathrm{proj}_{S^+}(\mathcal{A})) \le \pi_S(\mathcal{A}) \le \pi_{S^-}(\mathcal{A})$,

(b) $\delta(\mathcal{A}) \le \pi_S(\mathcal{A})\mu_{V \setminus S}(\mathcal{A})$,

(c) $\delta(\mathcal{A} \bowtie \mathcal{B}) \le \pi_S(\mathcal{A})\mu_{T \setminus S}(\mathrm{proj}_T(\mathcal{B}))\mu_{(V \cup W)\setminus(S \cup T)}(\mathcal{A} \bowtie \mathcal{B})$.

In particular, inequality (c)—bounding the density of a join—plays a critical role in our lower bound.

5. PATHSET COMPLEXITY

In this section, we define the key notion of pathset complexity, state our lower bound for pathset complexity (Theorem 3), and present a matching upper bound (Proposition 1).

Definition 4. (PATTERN GRAPH) Recall $P_k = (V_k, E_k)$ is the directed path of length k where $V_k = \{v_i : 0 \le i \le k\}$ and $E_k = \{v_i v_{i+1} : 0 \le i < k\}$. A *pattern graph* is a subgraph of P_k with no isolated vertices. That is, $G = (V_G, E_G)$ is a pattern graph if, and only if, $E_G \subseteq E_k$ and $V_G = \bigcup_{vw \in E_G} \{v, w\}$. We write $\wp_k$ for the set of pattern graphs (using power-set notation since pattern graphs are in 1-1 correspondence with subsets of E_k).

Note that every pattern graph is a (possibly empty) disjoint union of directed paths of length ≥ 1. We refer to maximal connected subsets of V_G simply as *components* of G. Two important parameters of pattern graphs are the number of components (= the number of maximal paths) and the length of the longest path (= the number of edges in the largest component). These are denoted by

$$\Delta_G := \#\text{ of components in } G \;(= |V_G| - |E_G|),$$
$$\ell_G := \text{length of the longest path in } G.$$

Definition 5. (PATHSET) For a pattern graph G, let $\mathscr{P}_G$ denote the power set of $[n]^{V_G}$. We refer to elements of $\mathscr{P}_G$ as *G-pathsets* (or just *pathsets* if G is clear from context).

The intuition for pathsets is as follows. For a pattern graph G, we view each $x \in [n]^{V_G}$ as corresponding to a "lifting" of G inside the complete layered graph $P_{k,n}$, namely isomorphic copy of G with vertex set $\{v^i \in V_{k,n} : i = x_v\}$ and edge set $\{v^i w^j \in E_{k,n} : i = x_v \text{ and } j = x_w\}$. In this view, a pathset $\mathcal{A} \subseteq [n]^{V_G}$ corresponds to a set of liftings of G. We have chosen to define *pathset* as a relation (a subset of $[n]^{V_G}$) rather than a set of liftings (which better matches intuition) in order to more naturally apply operations like $\bowtie$ and proj_S and μ_S, etc.

Definition 6. (G-SMALL PATHSETS)

(i) Let $\varepsilon := 1/\log k$ and $\tilde{n} := n^{1-\varepsilon}$.

(ii) A pathset $\mathcal{A} \in \mathscr{P}_G$ is *G-small* (we simply say *small* when G is understood from context) if, for all $1 \le t \le \Delta_G$ and $S \subseteq V_G$ such that S is the union of t components of G, $\mathcal{A}$ satisfies the density constraint $\mu_S(\mathcal{A}) \le \tilde{n}^{-t}$, that is, for all $y \in [n]^{V_G \setminus S}$,

$$n^{-|S|} \cdot |\{x \in \mathcal{A} : x_{V_G \setminus S} = y\}| \le \tilde{n}^{-t}.$$

(iii) The set of G-small pathsets is denoted $\mathscr{P}_G^{\text{small}}$.

Remark 2. Clarifying this definition:

- As the terminology suggests, G-smallness is a monotone decreasing property (i.e. if $\mathcal{A}$ is G-small, then so is every $\mathcal{A}' \subseteq \mathcal{A}$).
- G-smallness consists of $2^{\Delta_G} - 1$ density constraints on $\mathcal{A}$, corresponding to the nonempty unions of the Δ_G components of G. Note that for $t = \Delta_G$ and $S = V_G$, the constraint $\mu_S(\mathcal{A}) \le \tilde{n}^{-t}$ is equivalent to $\delta(\mathcal{A}) \le \tilde{n}^{-\Delta_G}$. In the special case that G is connected (i.e. $\Delta_G = 1$), $\mathcal{A}$ is G-small $\iff$ $\delta(\mathcal{A}) \le \tilde{n}^{-1}$.
- The precise value of ε is not important: any ε between $1/k$ and $1/2$ would suit our purposes, modulo a slight weakening in the parameters of our main theorem.

Example 1. Let G be the pattern graph with components $U = \{v_1, v_2, v_3\}$ and $U' = \{v_5, v_6\}$ (i.e. $V_G = \{v_1, v_2, v_3, v_5, v_6\}$ and $E_G = \{v_1v_2, v_2v_3, v_5v_6\}$). A pattern $\mathcal{A} \in \mathscr{P}_G$ is G-small if, and only if,

$$\delta(\mathcal{A}) \le \tilde{n}^{-2}, \quad \mu_U(\mathcal{A}) \le \tilde{n}^{-1}, \quad \mu_{U'}(\mathcal{A}) \le \tilde{n}^{-1}.$$

For example, the pathset $\mathcal{A}_1 := \{x : x_1 = x_5 = 1\}$ is G-small (here x ranges over $[n]^{V_G}$ and we write x_i for x_{v_i}) since $\delta(\mathcal{A}_1) = n^{-2} < \tilde{n}^{-2}$ and $\mu_U(\mathcal{A}_1) = \mu_{U'}(\mathcal{A}_1) = n^{-1} < \tilde{n}^{-1}$. The pathset $\mathcal{A}_2 := \{x : x_1 = x_5 \text{ and } x_2 = x_6\}$ is G-small as well since $\delta(\mathcal{A}_2) = \mu_U(\mathcal{A}_2) = \mu_{U'}(\mathcal{A}_2) = n^{-2}$. On the other hand, pathsets

$$\mathcal{A}_3 := \{x : x_1 = x_2 = 1\}, \quad \mathcal{A}_4 := \{x : x_1 = x_5\}$$

are not G-small since $\mu_{U'}(\mathcal{A}_3) = 1 > \tilde{n}^{-1}$ and $\delta(\mathcal{A}_4) = n^{-1} > \tilde{n}^{-2}$.

The next lemma shows that smallness is preserved under joins. (See full paper for the proof.)

LEMMA 2. *If $\mathcal{A}$ is a small G-pathset and $\mathcal{B}$ is a small H-pathset, then $\mathcal{A} \bowtie \mathcal{B}$ is a small $G \cup H$-pathset.* $\square$

We now come to the key definition of *pathset complexity.*

Definition 7. (PATHSET COMPLEXITY) For every pattern graph G and pathset $\mathcal{A} \in \mathscr{P}_G$, the *pathset complexity* $\chi_G(\mathcal{A})$ of $\mathcal{A}$ with respect to G is defined by the following induction:

(i) If G is the empty graph, then $\chi_G(\mathcal{A}) := 0$.

(ii) If G consists of a single edge, then $\chi_G(\mathcal{A}) := |\mathcal{A}|$.

(iii) If G has ≥ 2 edges, then

$$\chi_G(\mathcal{A}) := \min_{(H_i, K_i, \mathcal{B}_i, \mathcal{C}_i)_i} \sum_i \chi_{H_i}(\mathcal{B}_i) + \chi_{K_i}(\mathcal{C}_i)$$

where $(H_i, K_i, \mathcal{B}_i, \mathcal{C}_i)_i$ ranges over sequences[4] where $H_i, K_i \subset G$, $H_i \cup K_i = G$, $\mathcal{B}_i \in \mathscr{P}_{H_i}^{\text{small}}$, $\mathcal{C}_i \in \mathscr{P}_{K_i}^{\text{small}}$ and $\mathcal{A} \subseteq \bigcup_i \mathcal{B}_i \bowtie \mathcal{C}_i$.

In plain language, we consider ***coverings*** of $\mathcal{A}$ by ***joins*** of ***small*** pathsets over ***proper*** subgraphs of G. The pathset complexity $\chi_G(\mathcal{A})$ is the minimum possible value—over all such coverings—of the sum of pathset complexities of the constituent small pathsets.

It is easily seen that pathset complexity satisfies the following inequalities:

(base case) $\chi_\emptyset(\{()\}) \le 0$ and $\chi_G(\mathcal{A}) \le 1$ if $|E_G| = |\mathcal{A}| = 1$,

(monotonicity) if $\mathcal{A}' \subseteq \mathcal{A}$, then $\chi_G(\mathcal{A}') \le \chi_G(\mathcal{A})$,

(sub-additivity) $\chi_G(\mathcal{A}_1 \cup \mathcal{A}_2) \le \chi_G(\mathcal{A}_1) + \chi_G(\mathcal{A}_2)$,

(join rule) if $\mathcal{A} \in \mathscr{P}_G^{\text{small}}$ and $\mathcal{B} \in \mathscr{P}_H^{\text{small}}$, then

$$\chi_{G \cup H}(\mathcal{A} \bowtie \mathcal{B}) \le \chi_G(\mathcal{A}) + \chi_H(\mathcal{B}).$$

We refer to these inequalities repeatedly throughout the paper.

Remark 3. Pathset complexity has a *dual characterization* as the unique pointwise maximal function from pairs $(G, \mathcal{A})$ to $\mathbb{R}$ which satisfies (base case), (monotonicity), (sub-additivity) and (join rule). This fact is established by a straightforward induction on pattern graphs. It gives an alternative view of pathset complexity as a "minimal construction cost", which is useful for proving upper bounds (see the full paper).

This dual characterization also suggests an obvious "direct method" for proving a lower bound on $\bar{\chi}$: find an explicit function from pairs $(G, \mathcal{A})$ to $\mathbb{R}$ and show that this function satisfies inequalities (base case), (monotonicity), (sub-additivity) and (join rule). This is analogous to proving a formula-size lower bound via a formal complexity measure.[5] We emphasize that our pathset complexity lower bound (Theorem 3) is not proved via the direct method; rather, our proof involves a more subtle induction (on the predominant *pattern* of joins; see the full paper for a detailed explanation).

We now state our lower bound on pathset complexity. (The proof of Theorem 3, which constitutes half of the full paper, is omitted in this abstract.)

[4]Without loss of generality, i ranges over $\mathbb{N}$ since $H_i = K_i = G$ and $\mathcal{B}_i = \mathcal{C}_i = \emptyset$ can occur infinitely often.

[5]A formal complexity measure is a function M from {boolean functions on n variables} to $\mathbb{R}$ satisfying inequalities $M(f \wedge g) \le M(f) + M(g)$ and $M(f \vee g) \le M(f) + M(g)$ in addition to base case inequalities $M(f) \le 0$ if f is constant and $M(f) \le 1$ if f is a coordinate function.

Theorem 3 (Pathset Complexity Lower Bound). *For all $\mathcal{A} \in \mathscr{P}_{P_k}$,*

$$\chi_{P_k}(\mathcal{A}) \geq 2^{-O(2^k)} \cdot n^{(1/6)\log k} \cdot \delta(\mathcal{A}).$$

For $k \leq \log\log n$ and non-negligible $\delta(\mathcal{A}) = n^{-O(1)}$, Theorem 3 implies $\chi_{P_k}(\mathcal{A}) \geq n^{(1/6)\log k - O(1)}$. We now give an upper bound which establishes that Theorem 3 is tight when $k \leq \log\log n$ and $\delta(\mathcal{A}) = n^{-O(1)}$.

Proposition 1 (Upper Bound). *For all $\mathcal{A} \in \mathscr{P}_{P_k}$,*

$$\chi_{P_k}(\mathcal{A}) \leq O(kn^{(1/2)\lceil \log k\rceil+2}).$$

For $k \leq \log\log n$ and $\mathcal{A} \in \mathscr{P}_{P_k}$ with $\delta(\mathcal{A}) = n^{-O(1)}$, our lower and upper bounds show that $\chi_{P_k}(\mathcal{A}) = n^{\Theta(\log k)}$ where the constant in $\Theta(\log k)$ is between $\frac{1}{6}$ and $\frac{1}{2}$.

Notation 1. For a pattern graph G and an integer s, we denote by $G^{\triangleright s}$ the *s-shifted* pattern graph with vertex set $\{v_{i+s} : v_i \in V_G\}$ and edge set $\{v_{i+s}v_{i+s+1} : v_iv_{i+1} \in E_G\}$. For a pathset $\mathcal{A} \in \mathscr{P}_G$, we denote by $\mathcal{A}^{\triangleright s} \in \mathscr{P}_{G^{\triangleright s}}$ the corresponding *s-shifted* pathset. Note that pathset complexity is invariant under shifts (i.e. $\chi_G(\mathcal{A}) = \chi_{G^{\triangleright s}}(\mathcal{A}^{\triangleright s})$).

Proof of Proposition 1. For simplicity we assume $\sqrt{n}$ is an integer. For all $k \geq 1$, define $\mathcal{A}_k \in \mathscr{P}^{\text{small}}_{P_k}$ by

$$\mathcal{A}_k := \{x \in [n]^{\{0,\dots,k\}} : x_0, x_k \leq \sqrt{n}\}.$$

(Note that $\delta(\mathcal{A}_k) = 1/n < 1/\tilde{n}$, so $\mathcal{A}_k$ is indeed P_k-small.)

Letting $j = \lceil k/2 \rceil$, we have

$$\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j} = \{x \in [n]^{\{0,\dots,k\}} : x_0, x_j, x_k \leq \sqrt{n}\}.$$

Note that $\mathcal{A}_k$ is covered by $\sqrt{n}$ "copies" of $\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j}$ where, for $1 \leq t \leq \sqrt{n}$,

$$\begin{aligned}&\text{Copy}_t(\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j}) :=\\ &\{x \in [n]^{\{0,\dots,k\}} : x_0, x_k \leq \sqrt{n} \text{ and } (t-1)\sqrt{n} < x_j \leq t\sqrt{n}\}.\end{aligned}$$

Pathset complexity is clearly invariant under "copies" in this sense (i.e. χ_G is invariant under the action of coordinate-wise permutations of $[n]$ on $\mathscr{P}_G$). We have

$$\begin{aligned}&\chi_{P_k}(\text{Copy}_t(\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j}))\\
&= \chi_{P_k}(\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j}) && \text{(invariance under "copies")}\\
&= \chi_{P_j}(\mathcal{A}_j) + \chi_{P^{\triangleright j}_{k-j}}(\mathcal{A}^{\triangleright j}_{k-j}) && \text{(join rule)}\\
&= \chi_{P_j}(\mathcal{A}_j) + \chi_{P_{k-j}}(\mathcal{A}_{k-j}) && \text{(invariance under shifts).}\end{aligned}$$

Since $\mathcal{A}_k \subseteq \bigcup_{1\leq t\leq\sqrt{n}} \text{Copy}_t(\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j})$, sub-additivity of χ implies

$$\begin{aligned}\chi_{P_k}(\mathcal{A}_k) &\leq \sum_{1\leq t\leq\sqrt{n}} \chi_{P_k}(\text{Copy}_t(\mathcal{A}_j \bowtie \mathcal{A}^{\triangleright j}_{k-j}))\\ &= \sqrt{n}\cdot\big(\chi_{P_j}(\mathcal{A}_j) + \chi_{P_{k-j}}(\mathcal{A}_{k-j})\big).\end{aligned}$$

This recurrence implies

$$\chi_{P_k}(\mathcal{A}_k) \leq (2\sqrt{n})^{\lceil \log k\rceil} \cdot \chi_{P_1}(\mathcal{A}_1) = O(kn^{(1/2)\lceil\log k\rceil+1}).$$

Now note that the complete P_k-pathset $[n]^{V_k}$ is covered by n "copies" of P_k. Therefore, by a similar argument,

$$\chi_{P_k}([n]^{V_k}) \leq n \cdot \chi_{P_k}(\mathcal{A}) = O(kn^{(1/2)\lceil\log k\rceil+2}).$$

Finally, by monotonicity, $\chi_{P_k}(\mathcal{A}) \leq O(kn^{(1/2)\lceil\log k\rceil+2})$ for all $\mathcal{A} \in \mathscr{P}_{P_k}$. $\square$

Remark 4. Note the importance of *smallness* in Proposition 1: if we relax the (join rule) inequality so that $\chi_{G\cup H}(\mathcal{A} \bowtie \mathcal{B}) \leq \chi_G(\mathcal{A}) + \chi_H(\mathcal{B})$ for *arbitrary* $\mathcal{A} \in \mathscr{P}_G$ and $\mathcal{B} \in \mathscr{P}_H$, then we could construct the complete P_k-pathset $[n]^{V_k}$ at a total cost of kn^2 simply by joining pathsets $[n]^{\{v_i,v_{i+1}\}}$ for $0 \leq i < k$. This shows that the smallness constraint on joins is essential to Theorem 3. Intuitively, smallness is responsible for bottlenecks which drive up the cost of constructing sufficiently dense pathsets. However, note that smallness is not an obstacle for constructing very sparse pathsets like $[\sqrt{n}]^{P_k}$: since pathsets $[\sqrt{n}]^{\{v_i,v_{i+1}\}}$ are small, we can take joins and show $\chi_{P_k}([\sqrt{n}]^{P_k}) \leq kn$.

6. FROM FORMULAS TO PATHSET COMPLEXITY

In this section we derive our main result (Theorem 2) from our lower bound on pathset complexity (Theorem 3). Let F_0 be a formula of depth $d(n)$ which solves *DISTCONN*(k,n) where $k(n) \leq \log\log n$ and $d(n) \leq \log n/k^3 \log\log n$. We must show that F_0 has size $n^{\Omega(\log k)}$.

As a first preliminary step: without loss of generality, we assume that F_0 has minimal size among all depth $d(n)$ formulas solving *DISTCONN*(k,n). In particular, we have $\text{size}(F_0) \leq kn^{k-1}$ since *DISTCONN*(k,n) has DNFs of this size.

As a second preliminary step, we convert F_0 into a fan-in 2 formula F by replacing each unbounded fan-in AND/OR gate by a balanced binary tree of fan-in 2 AND/OR gates. We have $\text{size}(F) = \text{size}(F_0) \leq n^k$ and

$$\text{depth}(F) \leq \text{depth}(F_0)\cdot\log(\text{size}(F_0)) \leq \log^2 n.$$

We write F_{in} for the set of inputs (i.e. leaves) in F, and F_{gate} for the set of gates in F, and f_{out} for the output gate in F. Note that each $f \in F$ is computed by an (unbounded fan-in) formula of size $\leq n^k$ and depth $\leq d(n)$ (by collapsing all adjacent AND/OR gates below f).

In order to lower bound size(F) in terms of pathset complexity, we define a family of pathsets $\mathcal{A}^{\Gamma}_{f,G}$ associated with each $f \in F$ and $G \in \wp_k$ and $\Gamma \in \{0,1\}^N$. Recall that we identify $\{0,1\}^N$ with the set of k-layered graphs where $N = E_{k,n} = \{v^iw^j : vw \in E_k,\ i,j \in [n]\}$.

Definition 8. (Pathsets $\mathcal{A}^{\Gamma}_{f,G}$) For all $G \in \wp_k$ and $x \in [n]^{V_G}$ and $\Gamma \in \{0,1\}^N$ and $f \in F$:

(i) Let $N_{G,x} := \{v^iw^j \in N : i = x_v \text{ and } j = x_w\}$ ($= \{v^{x_v}w^{x_w} : vw \in E_G\}$).

(ii) Let $\rho^{\Gamma}_{G,x} : N \to \{0,1,*\}$ be the restriction which equals $*$ over $N_{G,x}$ and agrees with Γ over $N \setminus N_{G,x}$. In particular, applying $\rho^{\Gamma}_{G,x}$ to f, we get a function $f\lceil\rho^{\Gamma}_{G,x} : \{0,1\}^{N_{G,x}} \to \{0,1\}$ (whose variables correspond to edges of G via the bijection $N_{G,x} \cong E_G$).

(iii) Let $\mathcal{A}^{\Gamma}_{f,G}$ be the G-pathset defined by

$$\mathcal{A}^{\Gamma}_{f,G} := \{x \in [n]^{V_G} : \mathit{Live}(f\lceil\rho^{\Gamma}_{G,x}) = N_{G,x}\}.$$

That is, $\mathcal{A}^{\Gamma}_{f,G}$ is the set of $x \in [n]^{V_G}$ such that the restricted function $f\lceil\rho^{\Gamma}_{G,x}$ depends on all $|N_{G,x}|$ ($= |E_G|$) of its variables.

In the next three subsections, we prove a sequence of claims about pathsets $\mathcal{A}^{\Gamma}_{f,G}$ in three cases where $f \in F_{\text{in}}$ and $f \in F_{\text{gate}}$ and $f = f_{\text{out}}$.

Remark 5. Claims 1, 2, 3 rely on few assumptions about F. In particular, these claims do not depend on the assumption that F_0 has bounded depth (i.e. F has bounded alternations), nor even that F is a formula as opposed to a circuit. In fact, these claims are valid if F is any B_2-circuit computing $\mathit{DISTCONN}(k,n)$ where B_2 is the full binary basis.

Of course, we will eventually use both assumptions that (I) F_0 has bounded depth (i.e. F has bounded alternations), and (II) F is a formula as opposed to a circuit. Our main technical lemma (Lemma 3) relies on (I) but not (II) (not surprisingly, since the proof uses the Switching Lemma, which does not distinguish between circuits and formulas). A second key lemma (Lemma 4) relies on (II) but not (I) (using a novel top-down argument which only works for formulas).

6.1 Inputs of F

Suppose f is an input in F labeled by a literal (i.e. a variable or its negation) corresponding to some $v^i w^j \in N$. Then we have the following explicit description of $\mathcal{A}^\Gamma_{f,G}$:

- if G is the empty graph, then $\mathcal{A}^\Gamma_{f,G} = \{()\}$ (i.e. the singleton containing the 0-tuple),
- if $E_G = \{vw\}$, then $\mathcal{A}^\Gamma_{f,G} = \{x\}$ for the unique $x \in [n]^{\{v,w\}}$ with $x_v = i$ and $x_w = j$,
- otherwise (i.e. if $|E_G| \geq 2$), $\mathcal{A}^\Gamma_{f,G} = \emptyset$.

By the base case conditions (i) and (ii) in Definition 7 of pathset complexity, we have $\chi_\emptyset(\mathcal{A}) = 0$ and $\chi_G(\mathcal{A}) = |\mathcal{A}|$ if G has a single edge. The upshot of these observations is the following claim.

Claim 1. *For all $f \in F_{\mathrm{in}}$, $\sum_{G \in \wp_k} \chi_G(\mathcal{A}^\Gamma_{G,f}) = 1$.*

6.2 Gates of F

Suppose f is an AND or OR gate in F with children f_1 and f_2. Consider any $G \in \wp_k$ and $x \in \mathcal{A}^\Gamma_{f,G}$ (assuming $\mathcal{A}^\Gamma_{f,G}$ is nonempty). By definition of $\mathcal{A}^\Gamma_{f,G}$, the function $f\lceil\rho^\Gamma_{G,x} : \{0,1\}^{N_{G,x}} \to \{0,1\}$ depends on all variables in $N_{G,x}$. Since $f\lceil\rho^\Gamma_{G,x}$ is the AND or OR of functions $f_1\lceil\rho^\Gamma_{G,x}$ and $f_2\lceil\rho^\Gamma_{G,x}$, each variable in $N_{G,x}$ is a live variable for one or both $f_1\lceil\rho^\Gamma_{G,x}$ and $f_2\lceil\rho^\Gamma_{G,x}$.

Define sub-pattern graph $G_1 \subseteq G$ as follows: for each $vw \in E_G$, let vw be an edge in G_1 if and only if $v^{x_v}w^{x_w}$ ($\in N_{G,x}$) is a live variable for the function $f_1\lceil\rho^\Gamma_{G,x}$. Define $G_2 \subseteq G$ in the same way with respect to f_2. Since

$$\begin{aligned}&\{v^{x_v}w^{x_w} : vw \in E_G\} = N_{G,x} = \\ &\quad = \mathit{Live}(f\lceil\rho^\Gamma_{G,x}) = \mathit{Live}(f_1\lceil\rho^\Gamma_{G,x}) \cup \mathit{Live}(f_2\lceil\rho^\Gamma_{G,x}),\end{aligned}$$

it follows that $G_1 \cup G_2 = G$.

Let $y = x_{V_{G_1}}$ be the restriction of x ($\in [n]^{V_G}$) to coordinates in V_{G_1}. By definition of G_1, we have

- $v^{y_v}w^{y_w} = v^{x_v}w^{x_w} \in \mathit{Live}(f_1\lceil\rho^\Gamma_{G,x})$ for all $vw \in E_{G_1}$,
- $v^{x_v}w^{x_w} \notin \mathit{Live}(f_1\lceil\rho^\Gamma_{G,x})$ for all $vw \in E_G \setminus E_{G_1}$.

It follows that $\mathit{Live}(f_1\lceil\rho^\Gamma_{G_1,y}) = \mathit{Live}(f_1\lceil\rho^\Gamma_{G,x}) = N_{G_1,y}$, hence $y \in \mathcal{A}^\Gamma_{f_1,G_1}$. Similarly, for $z = x_{V_{G_2}}$, we have $z \in \mathcal{A}^\Gamma_{f_2,G_2}$. This shows that $x \in \mathcal{A}^\Gamma_{f_1,G_1} \bowtie \mathcal{A}^\Gamma_{f_2,G_2}$.

The observation may be succinctly expressed as

$$\mathcal{A}^\Gamma_{f,G} \subseteq \bigcup_{G_1,G_2 \subseteq G \,:\, G_1 \cup G_2 = G} \mathcal{A}^\Gamma_{f_1,G_1} \bowtie \mathcal{A}^\Gamma_{f_2,G_2}.$$

Splitting this union into the cases that $G_1 = G$ or $G_2 = G$ or $G_1, G_2 \subset G$, we have proved:

Claim 2 (Gates of F). *For every $f \in F_{\mathrm{gates}}$ with children f_1, f_2 and every $G \in \wp_k$,*

$$\mathcal{A}^\Gamma_{f,G} \subseteq \mathcal{A}^\Gamma_{f_1,G} \cup \mathcal{A}^\Gamma_{f_2,G} \cup \bigcup_{G_1,G_2 \subset G \,:\, G_1 \cup G_2 = G} \mathcal{A}^\Gamma_{f_1,G_1} \bowtie \mathcal{A}^\Gamma_{f_2,G_2}.$$

6.3 Output of F

We now use the fact that F computes $\mathit{DISTCONN}(k,n)$. Our previous Claims 1 and 2 applied to arbitrary $\Gamma \in \{0,1\}^N$. We now shift perspective and consider *random* $\Gamma \in \{0,1\}^N_{1/n}$. That is, Γ is the random k-layered graph (i.e. subgraph of $P_{k,n}$) with edge probability $1/n$. Recall that $V_{k,n} = \{v^i : v \in V_k \text{ and } i \in [n]\}$ and s,t are the vertices v^1_0, v^1_k. Each $x \in [n]^{V_k}$ corresponds to a path of length k in $P_{k,n}$, where x is an st-path if and only if $x_0 = x_k = 0$ (writing x_i instead of x_{v_i} for the coordinates of x).

Almost surely, Γ satisfies the following properties:

(i) Γ contains no st-path, and

(ii) all vertices in Γ have total degree (in-degree plus out-degree) $\leq \log^2 n$.

Both (i) and (ii) follow from simple union bounds. For (i), the number of st-paths is n^{k-1}, and each st-path only has probability n^{-k} of being in Γ. For (ii), the number of vertices is kn^2, and the probability of any given vertex having total degree $\geq \log^2 n$ is $\leq \binom{2n}{\log^2 n} n^{-\log^2 n} \leq (\frac{2e}{\log^2 n})^{\log^2 n} \leq n^{-\omega(1)}$.

For an st-path x, we will say that x is Γ-*independent* if Γ contains no path from x_i to x_j for all $0 \leq i < j \leq k$. We claim that, if Γ satisfies (i) and (ii), then 99% of st-paths are Γ-independent. To see this, consider the following greedy procedure for constructing a Γ-independent st-path. Sequentially, for $i = 1, \dots, k-1$, choose any x_i in the ith layer of $V_{k,n}$ such that Γ contains no path from s to x_i (this eliminates $\leq \log^{2i} n$ choices for x_i), nor a path from x_i to t (this eliminates $\leq \log^{2(k-i)} n$ choices), nor a path from $x_{i'}$ to x_i for any $1 \leq i' < i$ (this eliminates $\leq \sum_{i'=1}^{i-1} \log^{2(i-i')} n$ choices). Setting $x_0 = s$ and $x_k = t$, note that x is Γ-independent. In total we get $\geq (n - k^2 \log^{2k} n)^{k-1} \geq .99n^{k-1}$ distinct Γ-independent st-paths.

Suppose x is a Γ-independent st-path and let $e_1, \dots, e_k$ be the k edges in x. We claim that $\Gamma \cup \{e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_k\}$ contains no st-path for all $1 \leq i \leq k$. To see this, assume for the sake of contradiction that x' is an st-path in $\Gamma \cup \{e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_k\}$. Let $e'_1, \dots, e'_k$ be the edges of x'. Since e_i is a non-edge of Γ, we have $e_i \neq e'_i$. Starting at the endpoint of e'_i, we can follow the path x' forwards until reaching a vertex in x; we can also follow x' backwards from the initial vertex of e'_i until reaching a vertex in x. This segment of x' is a path in Γ between two vertices of x, contradiction Γ-independence of x.

Since f_{out} computes $\mathit{DISTCONN}(k,n)$, it follows that

$$f_{\mathrm{out}}(\Gamma \cup \{e_1, \dots, e_k\}) = 1,$$
$$f_{\mathrm{out}}(\Gamma \cup \{e_1, \dots, e_{i-1}, e_{i+1}, \dots, e_k\}) = 0 \text{ for all } 1 \leq i \leq k.$$

This shows that the restricted function $f_{\text{out}}\lceil\rho^{\Gamma}_{P_k,x}$ depends on all k unrestricted variables (corresponding to the edges of x); in fact, $f_{\text{out}}\lceil\rho^{\Gamma}_{P_k,x}$ is the AND function. Therefore, $x \in \mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}$ for every Γ-independent st-path x.

By this argument, we have proved:

Claim 3 (Output of F).

$$\lim_{n\to\infty} \mathbb{P}_{\Gamma\in\{0,1\}^N_{1/n}} [\,\delta(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \geq .99n^{-2}\,] = 1.$$

6.4 Proof of Theorem 2

We first present the two main lemmas in the reduction from formula size to pathset complexity. Lemma 3 is the main technical lemma and the only place in the overall proof of Theorem 3 which depends on the assumption that F has bounded depth; however, Lemma 3 does not depend on the fact that F is a formula as opposed to a circuit.

Lemma 3 (Pathsets $\mathcal{A}^{\Gamma}_{f,G}$ are Small). *Suppose $f : \{0,1\}^N \to \{0,1\}$ is computed by a circuit of depth $\leq \frac{\log n}{k^3 \log\log n}$ and size $\leq n^k$. Then, for all $G \in \wp_k$,*

$$\mathbb{P}_{\Gamma\in\{0,1\}^N_{1/n}} [\,\mathcal{A}^{\Gamma}_{f,G} \textit{ is not } G\textit{-small}\,] \leq O(n^{-2k}).$$

The proof of Lemma 3 is omitted in this extended abstract (see §7 of the full paper). The proof uses the technique developed in [19, 20], which relies on the Switching Lemma [6] and Janson's Inequality [9].

Lemma 4, below, is the nexus between formula size and pathset complexity. The proof involves a novel top-down argument, which is key to distinguishing formulas and circuits. (Though we will apply Lemma 4 to the formula F which we have been considering so far, Lemma 4 is stated in general terms for arbitrary boolean functions with fan-in 2.)

Lemma 4. *Let F be any fan-in 2 formula and let $\Gamma \in \{0,1\}^N$. If $\mathcal{A}^{\Gamma}_{f,G} \in \mathscr{P}^{\text{small}}_G$ for all $f \in F$ and $G \in \wp_k$, then*

$$\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \leq 2^{O(k^2)} \cdot \text{depth}(F)^k \cdot \text{size}(F).$$

Proof. Assume $\mathcal{A}^{\Gamma}_{f,G} \in \mathscr{P}^{\text{small}}_G$ for all $f \in F$ and $G \in \wp_k$. Consider any $f \in F_{\text{gates}}$ with children f_1 and f_2. By Claim 2, together with the key properties (monotonicity), (subadditivity) and (join rule) of pathset complexity, we have

$$\begin{aligned}\chi_G(\mathcal{A}^{\Gamma}_{f,G}) &\leq \chi_G\Big(\mathcal{A}^{\Gamma}_{f_1,G} \cup \mathcal{A}^{\Gamma}_{f_2,G} \cup \bigcup_{\substack{G_1,G_2\subset G:\\ G_1\cup G_2=G}} \mathcal{A}^{\Gamma}_{f_1,G_1} \bowtie \mathcal{A}^{\Gamma}_{f_2,G_2}\Big)\\ &\leq \chi_G(\mathcal{A}^{\Gamma}_{f_1,G}) + \chi_G(\mathcal{A}^{\Gamma}_{f_2,G})\\ &\quad + \sum_{\substack{G_1,G_2\subset G:\\ G_1\cup G_2=G}} \Big(\chi_{G_1}(\mathcal{A}^{\Gamma}_{f_1,G_1}) + \chi_{G_2}(\mathcal{A}^{\Gamma}_{f_2,G_2})\Big)\\ &\leq \sum_{i\in\{1,2\}} \Big(\chi_G(\mathcal{A}^{\Gamma}_{f_i,G}) + 2^k \sum_{H\subset G} \chi_H(\mathcal{A}^{\Gamma}_{f_i,H})\Big).\end{aligned}$$

Starting with $\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k})$ and repeatedly applying the above inequality until reaching the inputs of F, we get a bound of the form

$$\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \leq \sum_{f\in F_{\text{in}},\, G\in\wp_k} c_{f,G} \cdot \chi_G(\mathcal{A}^{\Gamma}_{f,G})$$

for some $c_{f,G} \in \mathbb{Z}_{\geq 0}$. We claim that

$$\begin{aligned}c_{f,G} &\leq \sum_{i,H_0,\dots,H_i\,:\,P_k=H_0\supset\cdots\supset H_i=G} 2^{ik} \cdot \binom{\text{depth of } f \text{ in } F}{i}\\ &\leq 2^{O(k^2)} \text{depth}(F)^k.\end{aligned}$$

To see this, consider any $f \in F_{\text{in}}$ and $G \in \wp_k$ and let $f_{\text{out}} = f_0, \dots, f_d = f$ be the branch in F from the output gate down to f. Then in the expansion of $\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k})$, we get a contribution of 2^{ik} ($\leq 2^{k^2}$) from each sequence $(i, t_0, H_0, t_1, H_1, \dots, t_i, H_i)$ where $0 = t_0 < \cdots < t_i = d$ and $P_k = H_0 \supset \cdots \supset H_i = G$; here t_i is the location where the expansion of $\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k})$ branches as we move from $\chi_{H_{i-1}}(\mathcal{A}^{\Gamma}_{f_{i-1},H_{i-1}})$ to $2^k\chi_{H_i}(\mathcal{A}^{\Gamma}_{f_i,H_i})$. Finally, we bound the number of $(t_0, \dots, t_i)$ by $\binom{d}{i}$ ($\leq \text{depth}(F)^k$) and the number of $(H_0, \dots, H_i)$ by 2^{ik} ($\leq 2^{k^2}$). Summing over i adds only a factor of k, so in total we get $c_{f,G} \leq 2^{O(k^2)}\text{depth}(F)^k$.

We now use the fact that $\sum_{G\in\wp_k} \chi_G(\mathcal{A}^{\Gamma}_{f,G}) = 1$ for all $f \in F_{\text{in}}$ (Claim 1) and $\text{size}(F) = |F_{\text{in}}|$ (since F is a formula!). Concluding the proof, we have

$$\chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \leq 2^{O(k^2)} \cdot \text{depth}(F)^k \cdot \text{size}(F). \qquad \square$$

Finally, we prove Theorem 2 assuming our pathset complexity lower bound (Theorem 3) and main technical lemma (Lemma 3).

Proof of Theorem 2. We must show that $\text{size}(F) \geq n^{\Omega(\log k)}$. By Claim 3 and Lemma 3, there exists $\Gamma \in \{0,1\}^N$ such that $\delta(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \geq .99n^{-2}$ and $\mathcal{A}^{\Gamma}_{f,G} \in \mathscr{P}^{\text{small}}_G$ for all $f \in F$ and $G \in \wp_k$. Fix any such Γ. By Lemma 4 and Theorem 3, we have

$$\begin{aligned}\text{size}(F) &\geq \frac{1}{2^{O(k^2)}\text{depth}(f)^k} \cdot \chi_{P_k}(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k})\\ &\geq \frac{1}{2^{O(k^2)}\text{depth}(f)^k} \cdot \frac{n^{(1/6)\log k}}{2^{O(2^k)}} \cdot \delta(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}).\end{aligned}$$

Using

$$\text{depth}(F) \leq \log^2 n,\;\; \delta(\mathcal{A}^{\Gamma}_{f_{\text{out}},P_k}) \geq .99n^{-2},\;\; k \leq \log\log n,$$

we get the desired bound $\text{size}(F) \geq n^{(1/6)\log k - O(1)}$. $\square$

7. CONCLUSION

We proved the first super-polynomial separation in the power of bounded-depth boolean formulas vs. circuits via technique based on the notion of pathset complexity. The most obvious question for future research is whether pathset complexity can be used to derive lower bounds for distance $k(n)$ connectivity in other models of computation.

We conclude with a comment extending our results to the *average-case* setting. Let $p(n) = \Theta(n^{-\frac{k+1}{k}})$ be the exact threshold function such that

$$\mathbb{P}_{G=G(n,p)} [\,G \in \mathit{STCONN}(k(n))\,] = 1/2$$

where $G(n,p)$ is the Erdős-Rényi random graph with edge probability $p(n)$. Our proof of Theorem 2 is easily adapted to give the same $n^{(1/6)\log k - O(1)}$ lower bound for bounded-depth formulas F which satisfy

$$\mathbb{P}_{G=G(n,p)} [\,F(G) = 1 \Leftrightarrow G \in \mathit{STCONN}(k(n))\,] \geq 1/2 + \varepsilon$$

for any constant $\varepsilon > 0$. Using the idea behind Proposition 1, we can construct formulas F of size $n^{(1/2)\log k + O(1)}$ (the best worst-case upper bound I know of is size $n^{\log k + O(1)}$) and depth $O(\log k)$ which solve $STCONN(k(n))$ in a strong average-case sense:

$$\mathop{\mathbb{P}}_{G = G(n,p)} [\, F(G) = 1 \Leftrightarrow G \in STCONN(k(n)) \,] \geq 1 - e^{-n^{\Omega(1)}}.$$

It would be interesting to close the gap between $\frac{1}{6}\log k$ and $\frac{1}{2}\log k$ in these bounds.

Acknowledgements

Thanks to Stasys Jukna, Igor Carboni Oliveira, Rahul Santhanam and Osamu Watanabe for valuable feedback, and to the anonymous referees for their helpful comments.

8. REFERENCES

[1] M. Ajtai. First-order definability on finite structures. *Annals of Pure and Applied Logic*, 45(3):211–225, 1989.

[2] P. Beame, R. Impagliazzo, and T. Pitassi. Improved depth lower bounds for small distance connectivity. *Computational Complexity*, 7(4):325–345, 1998.

[3] S. Bellantoni, T. Pitassi, and A. Urquhart. Approximation and small-depth frege proofs. *SIAM Journal on Computing*, 21(6):1161–1179, 1992.

[4] J. Edmonds, C. K. Poon, and D. Achlioptas. Tight lower bounds for st-connectivity on the NNJAG model. *SIAM Journal on Computing*, 28(6):2257–2284, 1999.

[5] M. L. Furst, J. B. Saxe, and M. Sipser. Parity, circuits, and the polynomial-time hierarchy. *Mathematical Systems Theory*, 17:13–27, 1984.

[6] J. Håstad. *Computational limitations of small-depth circuits*. MIT press, 1987.

[7] J. Håstad. The shrinkage exponent of de Morgan formulas is 2. *SIAM Journal on Computing*, 27(1):48–64, 1998.

[8] J. Håstad, S. Jukna, and P. Pudlák. Top-down lower bounds for depth-three circuits. *Computational Complexity*, 5(2):99–112, 1995.

[9] S. Janson. Poisson approximation for large deviations. *Random Structures & Algorithms*, 1(2):221–229, 1990.

[10] S. Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27. Springer-Verlag Berlin Heidelberg, 2012.

[11] M. Karchmer and A. Wigderson. Monotone circuits for connectivity require super-logarithmic depth. *SIAM Journal on Discrete Mathematics*, 3(2):255–265, 1990.

[12] N. Nisan and A. Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Computational Complexity*, 6(3):217–234, 1996.

[13] A. Potechin. Bounds on monotone switching networks for directed connectivity. In *Foundations of Computer Science (FOCS), 2010 51st Annual IEEE Symposium on*, pages 553–562. IEEE, 2010.

[14] R. Raz and A. Wigderson. Probabilistic communication complexity of boolean relations. In *Foundations of Computer Science, 1989., 30th Annual Symposium on*, pages 562–567. IEEE, 1989.

[15] A. A. Razborov. Lower bounds on the size of bounded depth circuits over a complete basis with logical addition. *Math. Notes*, 41:333–338, 1987.

[16] A. A. Razborov and S. Rudich. Natural proofs. *J. Comput. Syst. Sci.*, 55(1):24–35, 1997.

[17] O. Reingold. Undirected connectivity in log-space. *Journal of the ACM (JACM)*, 55(4):17, 2008.

[18] J. Riordan and C. E. Shannon. The number of two-terminal series-parallel networks. *J. Math. Phys*, 21(2):83–93, 1942.

[19] B. Rossman. On the constant-depth complexity of k-clique. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pages 721–730. ACM, 2008.

[20] B. Rossman. *Average-case complexity of detecting cliques*. PhD thesis, Massachusetts Institute of Technology, 2010.

[21] P. Savickỳ and A. R. Woods. The number of boolean functions computed by formulas of a given size. *Random Structures & Algorithms*, 13(3-4):349–382, 1998.

[22] W. J. Savitch. Relationships between nondeterministic and deterministic tape complexities. *Journal of computer and system sciences*, 4(2):177–192, 1970.

[23] E. Shamir and M. Snir. On the depth complexity of formulas. *Mathematical Systems Theory*, 13(1):301–322, 1979.

[24] R. Smolensky. Algebraic methods in the theory of lower bounds for boolean circuit complexity. In *STOC '87: Proceedings of the 19th Annual ACM Symposium on Theory of Computing*, pages 77–82, 1987.

[25] P. Spira. On time-hardware complexity tradeoffs for boolean functions. In *Proceedings of the 4th Hawaii Symposium on System Sciences*, pages 525–527, 1971.

[26] P. Tiwari and M. Tompa. A direct version of Shamir and Snir's lower bounds on monotone circuit depth. *Information Processing Letters*, 49(5):243–248, 1994.

[27] A. Wigderson. The complexity of graph connectivity. In *Proceedings of the 17th International Symposium on Mathematical Foundations of Computer Science*, pages 112–132. Springer-Verlag, 1992.