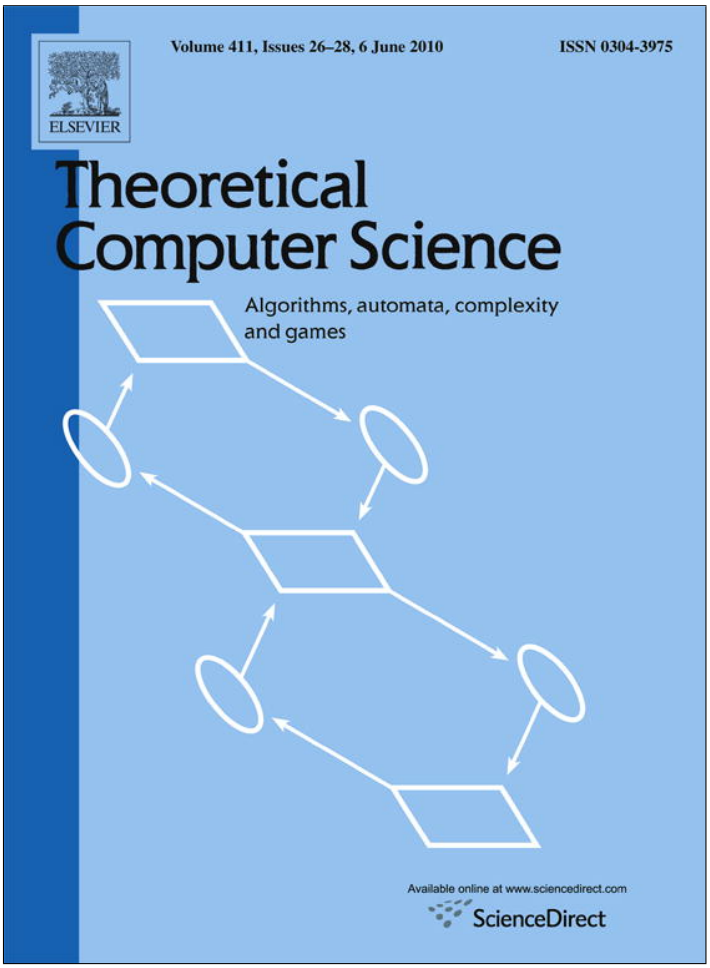

Theoretical Computer Science 411 (2010) 2612–2618

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

On the parity complexity measures of Boolean functions

Zhiqiang Zhang [a,*], Yaoyun Shi [b]

[a] Institute for Theoretical Computer Science, Center for Advanced Study, Tsinghua University, Beijing, 100084, PR China

[b] Department of Electrical Engineering and Computer Science, University of Michigan, 2260 Hayward Street, Ann Arbor, MI 48109-2121, USA

ARTICLE INFO

Article history:
Received 16 March 2009
Received in revised form 3 March 2010
Accepted 19 March 2010
Communicated by A. Razborov

Keywords:
Computational complexity
Communication complexity
Parity decision tree
Log-Rank conjecture

ABSTRACT

The parity decision tree model extends the decision tree model by allowing the computation of a parity function in one step. We prove that the deterministic parity decision tree complexity of any Boolean function is polynomially related to the non-deterministic complexity of the function or its complement. We also show that they are polynomially related to an analogue of the block sensitivity. We further study parity decision trees in their relations with an intermediate variant of the decision trees, as well as with communication complexity.

1. Introduction and summary of results

The decision tree model is perhaps the simplest model of computation. It is, however, capable of capturing the inherent complexity of many natural computational problems. Its relations with other models of computation have also proved to be useful. In this section, we will first review some definitions and key results on decision trees, before we present a summary of our results.

Let $f : \{0,1\}^n \to \{0,1\}$ be a Boolean function throughout this paper, unless specified otherwise. Formally, a decision tree algorithm for computing f is a full binary tree T, labeled as follows: (1) each non-leaf vertex is labeled with an index $i \in \{1,2,\ldots,n\}$ to the input bits, (2) each leaf and each edge is labeled with either 0 or 1. The computation of T on an input $x \in \{0,1\}^n$ is the path that starts at the root and follows the x_i edge from a vertex labeled with i. The leaf label reached by this path is the output of T on x. The depth of the tree is the worst-case complexity of the algorithm. The minimum depth of all decision trees computing f is the *deterministic decision tree complexity* of f, denoted by $D(f)$.

A set of decision trees *non-deterministically computes* f, if for any input x, $f(x) = 1$, if and only if a decision tree from the set outputs 1. The *non-deterministic decision tree complexity* of f, denoted by $C^1(f)$, is the smallest integer k such that f is computed non-deterministically by a set of depth-k decision trees. Alternatively, $C^1(f)$ is characterized by the smallest integer k, such that for any input x with $f(x) = 1$, there is a subset $S \subseteq \{1,\ldots,n\}$ such that any input x' with the same value as x on bits indexed by S must also have $f(x') = 1$. Thus $C^1(f)$ is also commonly called the 1-*certificate complexity*. The 0-certificate complexity, $C^0(f) \stackrel{\text{def}}{=} C^1(1-f)$, and the *certificate complexity*, $C(f) \stackrel{\text{def}}{=} \max\{C^0(f), C^1(f)\}$.

It follows straightforwardly from the definitions that $C(f) \le D(f)$. A key result [2] is, for any f,

$$D(f) \le C^1(f)C^0(f). \tag{1}$$

Thus for any Boolean function, its deterministic complexity is polynomially related with its non-deterministic complexity or that of its complement. This is in sharp contrast with the fact that for Turing machine computations the corresponding

* Corresponding author. Tel.: +86 13811567313.
E-mail addresses: zhang@itcs.tsinghua.edu.cn, mathzqy@gmail.com (Z. Zhang), shiyy@eecs.umich.edu (Y. Shi).

0304-3975/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2010.03.027

question of P versus NP remains open. In fact, several other complexity measures such as randomized and quantum decision tree complexities are also known to be polynomially related to the deterministic decision tree complexity. A comprehensive survey on the subject is [3] by Buhrman and de Wolf.

If in a decision tree, each non-leaf vertex is labeled with a $c \in \{0,1\}^n$ instead, and the computation path follows the edge labeled with $\langle x, c\rangle \stackrel{\text{def}}{=} \sum_i x_i c_i \bmod 2$, we call this extended decision tree a *parity decision tree* and the corresponding complexity as the *parity decision tree complexity*, denoted by $D_\oplus(f)$. This model was first defined in [4], which derived some simple properties of the complexity. The *parity certificate complexities*, $C^0_\oplus(f)$, $C^1_\oplus(f)$, and $C_\oplus(f)$, can be defined in analogy to the certificate complexities (see Definition 2.1). They measure the non-deterministic parity decision tree complexities of f (or $1-f$). Our first main result is in analogy to (1).

Theorem 1.1. *For any Boolean function f, $D_\oplus(f) \le C^0_\oplus(f)C^1_\oplus(f)$.*

The *block-sensitivity* of f, $\mathrm{bs}(f)$, is the smallest integer k such that for any input $x \in \{0,1\}^n$ there are k pair-wise disjoint subsets of $\{1,\ldots,n\}$ such that flipping all bits in any of those subsets flips $f(x)$. Nisan [8] showed that, for any f,

$$C(f) \le \mathrm{bs}^2(f). \tag{2}$$

Together with the simple relation that $\mathrm{bs}(f) \le C(f)$, this result shows that $\mathrm{bs}(f)$ is polynomially related with $C(f)$, thus with $D(f)$. We define (in Definition 3.3) the *parity block sensitivity* $\mathrm{bs}_\oplus(f)$, and show that a similar relation holds.

Theorem 1.2. *For any Boolean function f, $bs_\oplus(f) \le C_\oplus(f) \le bs^2_\oplus(f)$.*

The above three classes of parity complexities we study satisfy the following symmetry properties. Let $c \in \{0,1\}^n$. The function obtained by shifting f by c is $f_c : x \mapsto f(x+c)$. Let A be a linear transformation on $\{0,1\}^n$ (as the n-dimensional linear space over the field $\mathbb{F}_2$), f_A is the function defined as $f_A(x) = f(Ax)$. For any coset H of $\{0,1\}^n$ (i.e. a shift of a subspace), denote by $f|_H$ the restriction of f on H. A complexity measure Θ defined on Boolean functions is said to be invariant under shift if $\Theta(f_c) = \Theta(f)$ for any $c \in \{0,1\}^n$. It is said to be invariant under rotation if $\Theta(f_A) = T(f)$ for any invertible transformation A over $\mathbb{F}_2^n$.

When Θ is invariant under shift and rotation, we can extend the domain of Θ to include any function g defined on a coset H of $\{0,1\}^n$. For such a g, and a coset $H = c + S$ where $c \in \{0,1\}^n$ and S is a subspace with basis $\{e_1,\ldots,e_m\}$, we define $g' : \{0,1\}^m \to \{0,1\}$ as follows,

$$g'(x_1x_2\cdots x_m) \stackrel{\text{def}}{=} g(c + x_1e_1 + \cdots + x_me_m) \quad \text{for all } x \in \{0,1\}^m, \tag{3}$$

and extend Θ to g by setting,

$$\Theta(g) \stackrel{\text{def}}{=} \Theta(g'). \tag{4}$$

Then $\Theta(g)$ is well defined, as it is independent of the choice of the basis and c for H due to Θ being invariant under shift and rotation. We say a complexity measure Θ invariant under shift and rotation is *monotone* if for any $n \ge 1, f : \{0,1\}^n \to \{0,1\}$, and coset $H \subseteq \{0,1\}^n$, $\Theta(f|_H) \le \Theta(f)$.

All the classical complexity measures of Boolean functions such as decision tree complexity, certificate complexity, and block sensitivity are invariant only under shift but not under rotation. The parity version complexities we study are, however, invariant under both shift and rotation, and are monotone.

To contrast those two sets of complexity measures, we may "symmetrize" every classical complexity measure Θ to Θ_I by defining $\Theta_I(f) \stackrel{\text{def}}{=} \min_B \Theta(f_B)$, where B takes value from all invertible linear transformations. A natural question is if each parity complexity is identical, or at least polynomially related, to the rotation invariant version of the corresponding classical complexity. We show that this is not the case. In this sense, the parity decision tree model is an inherently more powerful model than the decision tree model.

Theorem 1.3. *For infinitely many n, there exists $f_n : \{0,1\}^n \to \{0,1\}$, such that $D_\oplus(f_n) = O(\log n)$ and $D_I(f_n) = \Theta(n)$.*

Parity decision trees are closely related to the communication complexity of XOR functions [10]. Communication complexity is a major branch of complexity theory that studies the inherent communication cost for distributive computation. The *deterministic communication complexity* of $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, denoted by $\mathrm{DC}(F)$, is the smallest integer k, such that there is a communication protocol between two parties Alice and Bob satisfying the following conditions: (1) Alice's input is an $x \in \{0,1\}^n$, and Bob's input is a $y \in \{0,1\}^n$. (2) Alice and Bob take turn to send each other a message, each message is determined by each party's input as well as the messages s/he has received previously. (3) At the end of the protocol one party knows $F(x,y)$. (4) The total number of bits in the messages is $\le k$. This model as well as its several variants have been extensively studied. For surveys, see [5,9,6].

Determining $\mathrm{DC}(F)$ may be a highly nontrivial problem, even for the following class of functions of a simple structure. A function $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ is called an XOR function [10] if for some $f : \{0,1\}^n \to \{0,1\}$, $F(x,y) = f(x+y)$, for all $x, y \in \{0,1\}^n$. The computation of a parity decision tree T for f can be simulated by Alice and Bob for computing F: each query c is simulated by Alice and Bob computing $\langle c, x\rangle$ and $\langle c, y\rangle$, respectively, and exchange the outcomes.

Proposition 1.4. *For any XOR function $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$ with $F(x,y) = f(x+y)$, $DC(F) \le 2D_\oplus(f)$.*

In Section 5, we show that $C^1(f)$, times $\log n$, also gives an upper bound on the *non-deterministic communication complexity* of F. A natural question is if those upper bounds are far from being tight. While we are not able to answer this question, we conjecture they are. We also put forward a conjecture that, if true, would also imply the well-known Log-Rank Conjecture [7] when restricted to XOR functions.

2. Parity certificate complexity

We consider $\{0,1\}^n$ as a n-dimensional vector space over $\mathbb{F}_2$, the two-element finite field, as well as an Abelian group with respect to the bit-wise XOR. Then a coset of $\{0,1\}^n$ is a set $b+V$, where $b \in \{0,1\}^n$ and V is a subspace of $\{0,1\}^n$. The co-dimension of $b+V$ is $n - \dim(V)$. Equivalently, a coset is the set of solutions to a system of linear equations, and the minimum number of the equations defining the same coset is the co-dimension. Informally, the parity certificate complexity measures how many linear constraints have to be given on the input in order to fix the value of f.

Definition 2.1. Let $f : D \to \{0,1\}$ be defined on $D \subseteq \{0,1\}^n$, and $x \in D$. A coset S of $\{0,1\}^n$ is called a *parity certificate* of f on x if $s \in S$ and f is constant on $S \cap D$. The *size* of the certificate is defined to be the co-dimension of S. The minimum size of a parity certificate for x is denoted by $C_\oplus(f,x)$. The *parity certificate complexity of* f, denoted by $C_\oplus(f)$, is $\max_x C_\oplus(f,x)$.

A parity certificate S is called a 0- (or 1-) parity certificate if $f(x) = 0$ (or $f(x) = 1$, respectively) for all $x \in S \cap D$. The 0- *and* 1-*parity certificate complexities of* f are $C^0_\oplus(f) \stackrel{\text{def}}{=} \max_{x:f(x)=0} C_\oplus(f,x)$, and $C^1_\oplus(f) \stackrel{\text{def}}{=} \max_{x:f(x)=1} C_\oplus(f,x)$, respectively.

If $f \equiv 0$ (or $f \equiv 1$), then $C^1_\oplus(f)$ (or $C^0_\oplus(f)$, respectively) is not defined. We may represent a parity certificate S of size T (or a coset S of co-dimension T) by a pair (C,r), where $C \in \{0,1\}^{T\times n}$ and $r \in \{0,1\}^T$, such that $S = \{x : Cx = r\}$. It follows from the definitions that when $B \in \{0,1\}^{n\times n}$ takes value from invertible matrices,

$$C_\oplus(f,x) = \min_B C(f_B, B^{-1}x). \tag{5}$$

Similar relations between the 0- and 1-parity certificates/certificates also hold. Note that 0- and 1-parity certificate complexity measure the non-deterministic parity decision tree complexity of f and $1-f$, respectively, with the non-deterministic parity decision tree complexity defined in analogy to the non-deterministic decision tree complexity. Since any parity decision tree gives a certificate of size no more than the depth of the tree for any input, we have the following relation.

Proposition 2.2. *For any Boolean function* f, $C_\oplus(f) \le D_\oplus(f)$.

We now prove Theorem 1.1, which states that $D_\oplus(f) \le C^0_\oplus(f)C^1_\oplus(f)$, for any f.

Proof of Theorem 1.1. The idea of the proof is similar to that in [2] for proving Inequality (1). We give an algorithm that computes f using no more than $C^1_\oplus(f)C^0_\oplus(f)$ queries.

Fix an input x_0. For a sequence of cosets $(C_1,r_1), (C_2,r_2), \ldots$, define $V_i \stackrel{\text{def}}{=} \{x : C_j x = C_j x_0, j = 1,2,\ldots,i\}$ for $i \ge 1$ and $V_0 \stackrel{\text{def}}{=} \{0,1\}^n$. By definition, $V_0 \supseteq V_1 \supseteq V_2 \supseteq \cdots$. The algorithm will examine a sequence of 1-parity certificates, (C_1,r_1), $(C_2,r_2), \ldots$, that it constructs incrementally from an initially empty sequence. It proceeds as follows: For $i = 1,2,\ldots$, if $f|_{V_{i-1}}$ is constant, output that constant and terminate. Otherwise, extend the current sequence of 1-parity certificates with a new one (C_i,r_i) for $f|_{V_{i-1}}$ of the smallest size. Since $f|_{V_{i-1}}$ is not constant, such a 1-parity certificate exists. Query the rows in C_i. If the answers agree with r_i, return 1. Otherwise continue with i incremented by 1.

The algorithm clearly outputs the correct answer. Since restricting a function on a subset does not increase $C^1_\oplus$, at most $C^1_\oplus(f)$ queries are made in the ith iteration, for each i. We prove that $f|_{V_T}$ is constant for some $T \le C^0_\oplus(f)$. Assume otherwise and fix an $x'_0 \in V_T$ with $T = C^0_\oplus(f)$ and $f(x'_0) = 0$. We argue that for each i, $1 \le i \le T$,

$$C_\oplus(f|_{V_i}, x'_0) \le C_\oplus(f|_{V_{i-1}}, x'_0) - 1. \tag{6}$$

Fix a parity certificate (C,r) for $f|_{V_{i-1}}$ containing x'_0 and of the smallest size. Since the linear system $\{C_i x = r_i, Cx = r\}$ does not have a solution in V_{i-1} but the system $\{C_i x = r_i\}$ does (by the definition of (C_i,r_i) being a 1-parity certificate for $f|_{V_{i-1}}$, which is non-constant), the row space of C has a non-empty intersection with the space spanned by the rows of $C_1,\ldots,C_i$. Assume without loss of generality that the intersection is spanned by the first k rows, for some $k \ge 1$, in C (otherwise, apply an appropriate invertible matrix on both sides of $Cx = r$), and denote the sub-matrix of C and r containing those rows by C' and r', and the remaining portions by C'' and r''. Any $x \in V_i$ satisfying $C''x = r''$ must have $C'x = C'x_0 = C'x'_0 = r'$, thus $Cx = r$, implying $f(x) = 0$. Thus (C'',r'') is a parity certificate containing x'_0 for $f|_{V_i}$, and Eq. (6) holds. Consequently, $C_\oplus(f,x'_0) \ge T + C_\oplus(f|_{V_T}, x'_0) \ge T+1 > C^0_\oplus(f)$, a contradiction. Therefore $f|_{V_T}$ is constant for some $T \le C^1_\oplus(f)$, and the algorithm uses no more than $C^1_\oplus(f)C^0_\oplus(f)$ number of queries. □

3. Parity block sensitivity

Recall that the *block sensitivity of f on an input x*, $\mathrm{bs}(f,x)$, is the smallest integer k, such that there exist $S_1, S_2, \ldots, S_k \subseteq \{1,2,\ldots,n\}$ that are pair-wise disjoint, and for each i, $1 \le i \le k$, $f(x) \neq f(x^{S_i})$, where $x^{S_i} \in \{0,1\}^n$ is obtained from x by flipping each bit indexed by S_i. The *block sensitivity of f*, $\mathrm{bs}(f)$, is $\max_x \mathrm{bs}(f,x)$. We define the parity analogues of those concepts. First define weak parity block sensitivity $\mathrm{wbs}(f,x)$ similar to the definition of parity certificate complexity.

Definition 3.1. The *weak parity block sensitivity of f on x* is

$$\mathrm{wbs}_\oplus(f,x) \stackrel{\mathrm{def}}{=} \min_B \mathrm{bs}(f_B, B^{-1}x).$$

The *weak parity block sensitivity of f* is

$$\mathrm{wbs}_\oplus(f) \stackrel{\mathrm{def}}{=} \max_x \mathrm{wbs}_\oplus(f,x).$$

Note that $\mathrm{wbs}_\oplus(f)$ is invariant under shift and rotation, so we can extend it to functions defined on a coset through Eq. (4). The following example shows that $\mathrm{wbs}_\oplus(f)$ is not monotone.

Example 3.2. Consider $f(x_1,x_2,x_3) = x_1 \oplus (x_2 \vee x_3)$. For any input x, we can always choose a basis $\{e_1,e_2,e_3\}$ such that $f(x+e_i) = f(x)$, $i = 1,2,3$. For example, when $x = 011$ we can choose the basis $\{010, 001, 111\}$. For such bases, any sensitive block contains at least two base vectors. So there is at most one sensitive block, implying $\mathrm{wbs}_\oplus(f,x) \le 1$. But with $H = \{x : x_1 = 0\}$, $f|_H(x_2,x_3) = x_2 \vee x_3$. This is the OR function on two variables, of which the parity block sensitivity is 2 at 0. Thus for this f, $\mathrm{wbs}_\oplus(f) < \mathrm{wbs}_\oplus(f|_H)$.

We modify $\mathrm{wbs}_\oplus$ to a parity complexity measure by taking maximum over all restrictions to cosets. Then it will be invariant under shift and rotation, and is monotone.

Definition 3.3. For a Boolean function $f : \{0,1\}^n \to \{0,1\}$, its *parity block sensitivity*, $bs_\oplus(f)$, is

$$\mathrm{bs}_\oplus(f) = \max_H \mathrm{wbs}_\oplus(f|_H),$$

where H takes value from the cosets of $\{0,1\}^n$.

Similar to Inequality (2), Theorem 1.2 implies that the parity block sensitivity is polynomially related to parity certificate complexity. We give below the proof for the Theorem, which states that $\mathrm{bs}_\oplus(f) \le C_\oplus(f) \le \mathrm{bs}^2_\oplus(f)$ for any f. The proof idea is also similar to that for proving (2) in [8].

Proof of Theorem 1.2. Since $C_\oplus$ is monotone, to prove $\mathrm{bs}_\oplus(f) \le C_\oplus(f)$, it suffices to prove $\mathrm{wbs}_\oplus(f,x) \le C_\oplus(f)$, for any x. This follows straightforwardly from the definition, the relation between block sensitivity and certificate complexity, and Eq. (5):

$$\mathrm{wbs}_\oplus(f,x) = \min_B \mathrm{bs}(f_B, B^{-1}x) \le \min_B C(f_B, B^{-1}x) = C_\oplus(f,x).$$

We prove the second inequality by showing $C_\oplus(f) \le \mathrm{wbs}_\oplus(f)\mathrm{bs}_\oplus(f)$. Since the three quantities are both invariant under shift, we assume without loss of generality that $C_\oplus(f)$ is achieved at $x = 0$. Also assume without loss of generality that $f(0) = 0$. Since $C_\oplus(f,x) = C_\oplus(f_B, B^{-1}x)$ for any invertible B and any x, we can further assume without loss of generality that $b \stackrel{\mathrm{def}}{=} \mathrm{wbs}_\oplus(f,0) = \mathrm{bs}(f,0)$. Let $S_1, S_2, \ldots, S_b \subseteq \{1,2,\ldots,n\}$ be a collection of disjoint and minimal sets achieving $\mathrm{bs}(f,0)$. Consider $S = \{x : x_i = 0, i \in S_1 \cup S_2 \cup \cdots S_b\}$. Then S is a parity certificate for f, as otherwise there would be a block $S' \subseteq \left(\{1,\ldots,S\} - \bigcup_{i=1}^b S_i\right)$ such that $f(0^{S'}) = 1$, contradicting that $b = \mathrm{bs}(f,0)$.

Fix an i, $1 \le i \le b$. Let $m = |S_i|$ and $S_i = \{a_1, a_2, \ldots, a_m\}$. Consider $f|_{H_i}$, where $H_i \stackrel{\mathrm{def}}{=} \{x : x_j = 0, j \in \{1,2,\ldots,n\} - S_i\}$. Then $f|_{H_i} : \{0,1\}^m \to \{0,1\}$ and

$$f|_{H_i}(y) = f\left(\sum_{i=1}^m y_i e_{a_i}\right), \quad \text{for all } y \in \{0,1\}^m.$$

Since S_i is minimal, for any $S'_i \subseteq S_i$, $f(0^{S'_i}) = 1$ if and only if $S'_i = S_i$. Thus $f|_{H_i}(y)$ is the AND function on m variables. Therefore $\mathrm{wbs}_\oplus(f|_{H_i}) = m$. Consequently, $m \le \mathrm{bs}_\oplus(f)$. Thus $C_\oplus(f) = C_\oplus(f,0) \le \sum_{i=1}^b |S_i| \le \mathrm{wbs}_\oplus(f,0)\mathrm{bs}_\oplus(f)$, implying $C_\oplus(f) \le \mathrm{wbs}_\oplus(f)\mathrm{bs}_\oplus(f)$. □

4. The gap between parity measures and symmetrized classical measures

In this section, we prove Theorem 1.3, which states that for infinitely many n, there exists $f_n : \{0,1\}^n \to \{0,1\}$, such that $D_\oplus(f_n) = O(\log n)$ and $D_I(f_n) = \Theta(n)$. We will define the desired function f_n by a random parity decision tree of logarithmic depth, then show that there exists such a parity decision tree of which the function requires linear certificate complexity, thus linear decision tree complexity.

For $A \in \{0,1\}^{m\times n}$, $s \in \{0,1\}^n$, define

$$\tau_A(s) \stackrel{\text{def}}{=} \min\{|s+v| : v \in \text{row space of } A\}.$$

We will need the following lemma to lower bound the certificate complexity.

Lemma 4.1. *Let $f : \{0,1\}^n \to \{0,1\}$, $s \in \{0,1\}^n$ and $f(x) = \langle x, s\rangle$ for all x in a coset $H = (A, r)$. Then $C(f) \geq \tau_A(s)$. In particular, $D(f) \geq \tau_A(s)$.*

Proof. Choose an arbitrary $x_0 \in H$. Let $\ell \stackrel{\text{def}}{=} C(f, x_0) \leq C(f)$. Suppose that $E \in \{0,1\}^{\ell\times n}$ describes a certificate. That is, each row in E contains all 0 but a single 1, and all x' with $Ex' = Ex_0$ must have $f(x') = f(x_0)$.

Now consider two sets of equations on the unknown $y \in \{0,1\}^n$:

$$\begin{cases} Ey = Ex_0 \\ Ay = r \\ \langle s, y\rangle = \langle s, x_0\rangle \end{cases} \quad \text{and} \quad \begin{cases} Ey = Ex_0 \\ Ay = r \\ \langle s, y\rangle = 1 - \langle s, x_0\rangle. \end{cases}$$

The first set of equations has a solution (e.g. $y = x_0$) but not the second set, since all y satisfying $Ay = r$ must have $\langle s, y\rangle = \langle s, x_0\rangle$. This is possible only when s is in the span of the rows in E and in A. Thus for some v in the row space of A, $s + v$ is in the row space of E. Thus $\tau_A(s) \leq \ell$. Therefore, $\tau_A(s) \leq C(f)$. That $D(f) \geq \tau_A(s)$ follows from the fact that $C(f) \leq D(f)$. □

We are ready to prove Theorem 1.3.

Proof of Theorem 1.3. Let $n = 2^k$. We construct a function f with n variables decided by a parity decision tree T of depth $k + 4$. For $1 \leq i \leq k + 3$, all the i-th layer nodes are labeled by $e_i \stackrel{\text{def}}{=} 0^{i-1}10^{n-i}$. The t-th node of the last layer before the output, $1 \leq t \leq 8n$, is labeled by a random $s_t \in \{0,1\}^n$. The answer to this query $\langle x, s_t\rangle$ is the output.

Fix an invertible matrix B. Then f_B is computed by the parity tree that replaces each query c in T by $B^T c$. In this parity decision tree, the inputs that arrive at a node with query $s'_t \stackrel{\text{def}}{=} B^T s_t$ form a coset $H_t = (C_t, r_t)$ of co-dimension $k + 3$, and $f_B(x) = \langle x, s'_t\rangle$ for all $x \in H_t$. By Lemma 4.1, $D(f_B) \geq \tau_{C_t}(s'_t)$.

For each v in the row space of C_t, $s'_t + v$ is uniformly distributed. Thus by Hoeffding's Inequality, $\Pr(|s'_t + v| \leq n/4) \leq e^{-n/8}$. Thus

$$\Pr(\tau_{C_t}(s'_t) \leq n/4) \leq 2^{k+3}e^{-n/8} = 8ne^{-n/8}.$$

There are $8n$ independently chosen s_j, thus

$$\Pr(D(f_B) \geq n/4) \geq 1 - \left(8ne^{-n/8}\right)^{8n} = 1 - (8n)^{8n}e^{-n^2}.$$

There are at most $(2^n)^n = 2^{n^2}$ different transformations B (the exact number is $\Pi_{i=0}^{n-1}(2^{n-i} - 1)$). Therefore,

$$P(\min_B D(f_B) \geq n/4) \geq 1 - (8n)^{8n}e^{-n^2} \cdot 2^{n^2} = 1 - (8n)^{8n}\left(\frac{2}{e}\right)^{n^2} \to 1.$$

This implies that when n is large enough, almost all the functions f computed by the above parity trees have $D_I(f) = \min_B D(f_B) \geq n/4$. In contrast, the parity decision tree complexity of these f is no more than $k + 4 = \log_2 n + 4$. □

The following corollary follows from the polynomial relations among certificate complexity and block sensitivity with decision tree complexity and their analogy for parity complexities.

Corollary 4.2. *For infinitely many n, there exists a n-variate f_n such that the gaps between $C_\oplus(f)$ and $C_I(f)$ and between $bs_\oplus(f)$ and $bs_I(f)$ are exponential.*

5. Connection with communication complexities

In a *non-deterministic communication protocol* for computing $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}$, Alice or Bob may non-deterministically choose from a set of strategies for the rest of the communication. We say that the protocol computes F if for any (x, y), $F(x, y) = 1$ if and only if for some choice in the non-deterministic steps the protocol outputs 1. Denote the *non-deterministic communication complexity* of F by $N^1(F)$. A fundamental result by Aho, Ullman and Yannakakis [1] is $\mathrm{DC}(F) = O(N^1(F)N^1(1-F))$, a relation similar to those about decision tree complexity and parity decision tree complexity. The main result of this section relates $N^1(F)$ with $C^1_\oplus(f)$ for XOR functions F with $F(x, y) = f(x + y)$.

Theorem 5.1. *For any XOR function $F(x, y) = f(x \oplus y)$, $N^1(F) \le C^1_\oplus(f) \log n$.*

To prove this result, we will make use of the following notion.

Definition 5.2. A set $\mathcal{C}$ of 1-parity certificates for f is called *essential* if (1) for any x with $f(x) = 1$ there is an element in $\mathcal{C}$ containing x, (2) no element is a subset of the union of all the other elements, and (3) any element is of a size $C^1_\oplus(f)$.

Clearly there exists an essential set of 1-parity certificates, as one could start with one smallest 1-parity certificate for each x, increase its size to $C^1_\oplus(f)$ if necessary, and remove any element contained in the union of the rest of the set.

Proof of Theorem 5.1. Let $d = C^1_\oplus(f)$. Fix an essential set $\mathcal{C} = \{(C_i, r_i) : 1 \le i \le K\}$ of 1-parity certificates. The following is a simple non-deterministic communication protocol for F. bits of communication: Alice non-deterministically chooses $(C_i, r_i) \in \mathcal{C}$, sends i, as well as $C_i x$. Bob checks if $C_i x + C_i y = r_i$. He accepts if yes, rejects otherwise. The correctness of the protocol follows from the definition of 1-parity certificate and the assumption that $\mathcal{C}$ contains a 1-parity certificate for any 1-input. The total cost is $d + \lceil \log_2(K + 1) \rceil$. Lemma 5.3 shows that $K = n^{O(d)}$. Thus $N^1(F) = O(d \log n)$. □

Lemma 5.3. *Let $\mathcal{C}$ be an essential set of 1-parity certificates for $f : \{0,1\}^n \to \{0,1\}$ and $d = C^1_\oplus(f)$. Then $|\mathcal{C}| \le n^{O(d)}$.*

Proof. Let P be the number of pairs (x, C) that $x \in C$ and $C \in \mathcal{C}$. Since $|C| = 2^{n-d}$ for each C,

$$P = 2^{n-d} |\mathcal{C}|. \tag{7}$$

For each $x \in \{0,1\}^n$, let $S_1, S_2, \ldots, S_k \in \mathcal{C}$ be those that contains x. Then $V_i \stackrel{\text{def}}{=} x + S_i$, $1 \le i \le k$, are $n - d$-dimensional subspaces none of which is a subset of the union of the rest. We show below any such set of subspaces must have $k = n^{O(d)}$. Thus $P = 2^n n^{O(d)}$. Together with Eq. (7), this implies the conclusion that $|\mathcal{C}| = n^{O(d)}$.

Let $C_i \in \{0,1\}^{d \times n}$ such that $V_i = \{x : C_i x = 0\}$, $1 \le i \le k$. For any i, let $x_i \in V_i$ be such that $x_i \notin \bigcup_{j \ne i} V_j$. Then $C_i x_i = 0$, but $C_j x_i \ne 0$ for all $j \ne i$. Consider a $kd \times k$ matrix

$$G = \begin{bmatrix} C_1 \\ C_2 \\ \cdots \\ C_k \end{bmatrix} [x_1, x_2, \ldots, x_k].$$

Let rank_2 denote the rank over filed $\mathbb{F}_2$. Then $\mathrm{rank}_2(G) \le n$ from the above factorization of G. Represent G by a $k \times k$ block matrix a_{ij}, where each block a_{ij} is a $d \times 1$ vector.

For each t, $1 \le t \le d$, define the $k \times k$ submatrix $G^t = [a^t_{ij}]_{1 \le i,j \le k}$, where a^t_{ij} is the t-th element of a_{ij}. Since G^t is a submatrix of G, $\mathrm{rank}_2(G^t) \le \mathrm{rank}_2(G) \le n$.

Let $M = G^1 \vee G^2 \vee \cdots \vee G^d$ be the entry-wise conjunction of $G^1, G^2, \ldots, G^d$. Notice that for any matrix A and B, $A \vee B = A + B + A \odot B$, where $A \odot B$ is the entry-wise product of A and B. Since $\mathrm{rank}_2(A \odot B) \le \mathrm{rank}_2(A)\mathrm{rank}_2(B)$, we have

$$\mathrm{rank}_2(A \vee B) \le \mathrm{rank}_2(A) + \mathrm{rank}_2(B) + \mathrm{rank}_2(A \odot B) \le 3\mathrm{rank}_2(A)\mathrm{rank}_2(B).$$

Thus $\mathrm{rank}_2(M) < (3n)^d$. On the other hand, from the fact that $a_{ij} = 0$ iff $i = j$, $M = I - J$, where I is the identity matrix and J the all 1 matrix. Thus $\mathrm{rank}_2(M) \ge \mathrm{rank}_2(I) - \mathrm{rank}_2(J) = k - 1$. This implies $k = |\mathcal{V}| \le (3n)^d$. □

The following conjecture, if true, would imply that $\mathrm{DC}(F)$ is polynomially related to $D_\oplus(f)$ (as well as $C_\oplus(f)$), by the Aho–Ullman–Yannakakis Theorem and Theorem 1.1.

Conjecture 5.4. *For any XOR function F based on f, $N^1(F) = \Omega(C^1_\oplus(f))$.*

A major open problem on deterministic communication complexity is the Log-Rank Conjecture [7]. Denote by $\mathrm{rank}(F) = \mathrm{rank}([F(x, y)]_{x,y \in \{0,1\}^n})$, where $\mathrm{rank}(\cdot)$ is the rank over the reals. The Log-Rank Conjecture states that

$$\mathrm{DC}(F) = \log^{O(1)} \mathrm{rank}(F), \quad \text{for any } F. \tag{8}$$

The study of XOR functions is partly motivated by the Log-Rank Conjecture. Denote by

$$\|\hat{f}\|_0 = |\{\hat{f}_w \ne 0 : w \in \{0,1\}^n\}|,$$

where

$$\hat{f}_w = \frac{1}{2^n} \sum_{x \in \{0,1\}^n} (-1)^{\langle x, w \rangle} f(x)$$

is the Fourier coefficient of f on w. Then for any XOR function F based on f, $\text{rank}(F) = \|\hat{f}\|_0$. Our conjecture below, if true, would imply the Log-Rank Conjecture on XOR functions.

Conjecture 5.5. *For any Boolean function* $f : \{0, 1\}^n \to \{0, 1\}$, $D_{\oplus}(f)$ *and* $C_{\oplus}(f)$ *are polynomially related with* $\log \|\hat{f}\|_0$.

Acknowledgements

We thank Xiaoming Sun and Andrew Yao for helpful discussions. The first author was supported in part by the National Natural Science Foundation of China Grant 60553001, and the National Basic Research Program of China Grant 2007CB807900, 2007CB807901. The second author was supported in part by the National Science Foundation of the United States under the grants 0347078 and 0622033.

References

[1] A.V. Aho, J.D. Ullman, M. Yannakakis, On notions of information transfer in vlsi circuits, in: Proceedings of the Fifteenth Annual ACM Symposium on Theory of Computing, New York, NY, USA, 1983, pp. 133–139.
[2] R. Beals, H. Buhrman, R. Cleve, M. Mosca, R. de Wolf, Quantum lower bounds by polynomials, Journal of the ACM 48 (4) (2001) 778–797.
[3] H. Buhrman, R. de Wolf, Complexity measures and decision tree complexity: a survey, Theoretical Computer Science 288 (1) (2002) 21–43.
[4] E. Kushilevitz, Y. Mansour, Learning decision trees using the fourier spectrum, in: Proceedings of the Twenty-Third Annual ACM Symposium on Theory of Computing, ACM, New York, NY, USA, 1991, pp. 455–464.
[5] E. Kushilevitz, N. Nisan, Communication Complexity, Cambridge University Press, Cambridge, 1997.
[6] T. Lee, A. Shraibman, Lower bounds in communication complexity, Foundations and Trends in Theoretical Computer Science 3 (4) (2009) 263–398.
[7] L. Lovász, M. Saks, Lattices, mobius functions and communication complexity, in: 29th Annual Symposium on Foundations of Computer Science, FOCS '88, IEEE Computer Society Press, Los Angeles, CA, USA, 1988, pp. 81–90.
[8] N. Nisan, CREW PRAMs and decision trees, SIAM Journal on Computing 20 (6) (1991) 999–1007.
[9] A.A. Sherstov, Communication lower bounds using dual polynomials, Bulletin of the European Association for Theoretical Computer Science 95 (2008) 59–93.
[10] Z. Zhang, Y. Shi, Communication complexities of symmetric XOR functions, Quantum Information and Computation 9 (2009) 255–263.