Electronic Colloquium on Computational Complexity, Revision 1 of Report No. 99 (2012)

An improved lower bound for the randomized decision tree complexity of recursive majority

Nikos Leonardos

October 3, 2012

Abstract

We prove that the randomized decision tree complexity of the recursive majority-of-three is $\Omega(2.55^d)$, where d is the depth of the recursion. The proof is by a bottom up induction, which is same in spirit as the one in the proof of Saks and Wigderson in their FOCS 1986 paper on the complexity of evaluating game trees.

Previous work includes an $\Omega((7/3)^d)$ lower bound, presented in STOC 2003 by Jayram, Kumar, and Sivakumar. Their proof used a top down induction and tools from information theory. In ICALP 2011, Magniez, Nayak, Santha, and Xiao, improved the lower bound to $\Omega((5/2)^d)$ and the upper bound to $O(2.64946^d)$.

1 Introduction

In this paper we will be working with the decision tree model. We prove a lower bound on the randomized decision tree complexity of the recursive majority-of-three function.

Formally,

$$\mathrm{maj}_1(x_1, x_2, x_3) = \begin{cases} 1, & \text{if at least two of } x_1, x_2, x_3 \text{ are 1;} \\ 0, & \text{otherwise.} \end{cases}$$

$$\begin{aligned} \mathrm{maj}_{d+1}(x_1, \dots, x_{3^{d+1}}) = \mathrm{maj}_1(&\mathrm{maj}_d(x_1, \dots, x_{3^d}), \\ &\mathrm{maj}_d(x_{3^d+1}, \dots, x_{2\cdot 3^d}), \\ &\mathrm{maj}_d(x_{2\cdot 3^d+1}, \dots, x_{3^d})). \end{aligned}$$

We write maj for maj_1. The function can also be represented by a uniform ternary tree. In particular, let U_d be a tree of depth d, such that every internal node has three children and all leaves are on the same level. The function computed by interpreting U_d as a circuit with internal nodes labeled by maj-gates is maj_d.

This function seems to have been given by Ravi Boppana (see Example 1.2 in [7]) as an example of a function that has deterministic complexity 3^d, while its randomized complexity is asymptotically smaller. Other functions with this property are known. A notable example is the function

ISSN 1433-8092

nand_d, first analyzed by Snir [10]. This is the function represented by a uniform binary tree of depth d, with the internal nodes labeled by nand-gates. A simple randomized framework that can be used to compute both maj_d and nand_d is the following. Start at the root; as long as the output is not known, choose a child at random and evaluate it recursively. This type of algorithms are called in [7] *directional*. For maj_d the directional algorithm computes the output in $(8/3)^d$ queries. It was noted in [7] that better algorithms exist for maj_d. Interestingly, Saks and Wigderson show that the directional algorithm is optimal for the nand_d function, and show that its zero-error randomized decision tree complexity is $\Theta\big((\frac{1+\sqrt{33}}{4})^d\big)$. Their proof uses a bottom up induction and generalized costs. Their method of generalized costs allows them to charge for a query according to the value of the variable. Furthermore, they conjecture that the maximum gap between deterministic and randomized complexity is achieved for this function.

Inspired by their technique we prove an $\Omega\big(2.55^d\big)$ lower bound on maj_d that also holds for algorithms with bounded-error. (The bound of [7] for nand_d was extended to bounded-error algorithms by Santha in [8].) In contrast to the exact asymptotic bounds we have for nand_d, there had been no progress on the randomized decision tree complexity of maj_d for several years. However, recent papers have narrowed the gap between the upper and lower bounds for recursive majority. An $\Omega\big((7/3)^d\big)$ lower bound was showed in [4]. Jayram, Kumar, and Sivakumar, proved their bound using tools from information theory and a top down induction. Furthermore, they presented a non-directional algorithm that improves the $O\big((8/3)^d\big)$ upper bound. Magniez, Nayak, Santha, and Xiao [6], significantly improved the lower bound to $\Omega\big((5/2)^d\big)$ and the upper bound to $O\big(2.64946^d\big)$. (Both of these lower bounds hold for the case that the randomized decision tree is allowed to err.)

Our proof of the lower bound is simpler than the aforementioned ones; it doesn't require a background in information theory and it only uses induction. Note that, Landau, Nachmias, Peres, and Vanniasegaram [5], showed how to remove the information theoretic notions from the proof in [4], keeping its underlying structure the same. Our proof can be even more simplified, if one requires the known $\Omega\big(2.5^d\big)$ lower bound. A simpler proof of this bound seems to have been already known to Jonah Serman [9] in 2007.

We note that both functions maj_d and nand_d, belong to the class of *read-once functions*. These are functions that can be computed by read-once boolean formulae, that is, formulae with the property that each input variable appears exactly once. Heiman, Newman, and Wigderson [2] showed that read-once formulae with threshold gates have zero-error randomized complexity $\Omega(n/2^d)$ (here n is the number of variables and d the depth of a canonical tree-representation of the read-once function). Heiman and Wigderson [3] managed to show that for every read-once function f we have $R(f) \in \Omega\big(D(f)^{0.51}\big)$, where $R(f)$ and $D(f)$ are the randomized and deterministic complexity of f respectively. Note that the conjecture of Saks and Wigderson states that for every function f we have $R(f) \in \Omega\big(D(f)^{0.753\ldots}\big)$.

2 Definitions, notation, and preliminaries

In the following section we introduce basic concepts related to decision tree complexity. The reader can find a more complete exposition in the survey of Buhrman and de Wolf [1].

2.1 Definitions pertaining to decision trees

A *deterministic Boolean decision tree* Q over a set of variables $Z = \{z_i \mid i \in [n]\}$, where $[n] = \{1, 2, \ldots, n\}$, is a rooted and ordered binary tree. Each internal node is labeled by a variable $z_i \in Z$ and each leaf with a value from $\{0, 1\}$. An *assignment* to Z (or an *input* to Q) is a member of $\{0, 1\}^n$. The output $Q(\sigma)$ of Q on an input σ is defined recursively as follows. Start at the root and let its label be z_i. If $\sigma_i = 0$, we continue with the left child of the root; if $\sigma_i = 1$, we continue with the right child of the root. We continue recursively until we reach a leaf. We define $Q(\sigma)$ to be the label of that leaf. When we reach an internal node, we say that Q *queries* or *reads* the corresponding variable. We say that Q *computes* a Boolean function $f : \{0, 1\}^n \to \{0, 1\}$, if for all $\sigma \in \{0, 1\}^n$, $Q(\sigma) = f(\sigma)$. The *cost of Q on input σ*, $\mathrm{cost}(Q; \sigma)$, is the number of variables queried when the input is σ. The *cost of Q*, $\mathrm{cost}(Q)$, is its *depth*, the maximum distance of a leaf from the root. The *deterministic complexity*, $D(f)$, of a Boolean function f is the minimum cost of any Boolean decision tree that computes f.

A *randomized Boolean decision tree* Q_R is a distribution p over deterministic decision trees. On input σ, a deterministic decision tree is chosen according to p and evaluated. The *cost of Q_R on input σ* is $\mathrm{cost}(Q_R; \sigma) = \sum_Q p(Q) \,\mathrm{cost}(Q; \sigma)$. The *cost* of Q_R is $\max_\sigma \mathrm{cost}(Q_R; \sigma)$. A randomized decision tree Q_R *computes* a Boolean function f, if $p(Q) > 0$ only when Q computes f. A randomized decision tree Q_R *computes* a Boolean function f *with error δ*, if, for all inputs σ, $Q_R(\sigma) = f(\sigma)$ with probability at least $1 - \delta$. The *randomized complexity*, $R(f)$, of a Boolean function f is the minimum cost of any randomized Boolean decision tree that computes f. The *δ-error randomized complexity*, $R_\delta(f)$, of a Boolean function f, is the minimum cost of any randomized Boolean decision tree that computes f with error δ.

We are going to take a distributional view on randomized algorithms. Let μ be a distribution over $\{0, 1\}^n$ and Q_R a randomized decision tree. The *expected cost of Q_R under μ* is

$$\mathrm{cost}_\mu(Q_R) = \sum_\sigma \mu(\sigma) \,\mathrm{cost}(Q_R; \sigma).$$

The *δ-error expected complexity under μ*, $R^{\mu}_{\delta}(f)$, of a Boolean function f, is the minimum expected cost under μ of any randomized Boolean decision tree that computes f with error δ. Clearly, $R_{\delta}(f) \geq R^{\mu}_{\delta}(f)$, for any μ, and thus we can prove lower bounds on randomized complexity by providing lower bounds for the expected cost under any distribution.

2.2 Introducing cost-functions

Furthermore, we are going to utilize the method of generalized costs of Saks and Wigderson [7]. To that end, we define a *cost-function* relative to a variable set Z, to be a function $\phi : \{0,1\}^n \times Z \to \mathbb{R}$. We extend the previous cost-related definitions as follows. The cost of a decision tree Q under cost-function ϕ on input σ is

$$\mathrm{cost}(Q;\phi;\sigma) = \sum_{z \in S} \phi(\sigma;z),$$

where $S = \{z \,|\, z \text{ is queried by } Q \text{ on input } \sigma\}$. The cost of a randomized decision tree Q_R on input σ under cost-function ϕ is

$$\mathrm{cost}(Q_R;\phi;\sigma) = \sum_{Q} p(Q)\,\mathrm{cost}(Q;\phi;\sigma),$$

where p is the corresponding distribution over deterministic decision trees. Finally, the expected cost of a randomized decision tree Q_R under cost-function ϕ and distribution μ is

$$\mathrm{cost}_{\mu}(Q_R;\phi) = \sum_{\sigma} \mu(\sigma)\,\mathrm{cost}(Q_R;\phi;\sigma).$$

Observation 1. *Let ϕ and ψ be two cost-functions relative to Z. For any decision tree Q over Z, any assignment σ to Z, and any $a, b \in \mathbb{R}$, we have*

$$a\,\mathrm{cost}(Q;\phi;\sigma) + b\,\mathrm{cost}(Q;\psi;\sigma) = \mathrm{cost}(Q;a\phi + b\psi;\sigma).$$

Thus, for any distribution μ,

$$a\,\mathrm{cost}_{\mu}(Q;\phi) + b\,\mathrm{cost}_{\mu}(Q;\psi) = \mathrm{cost}_{\mu}(Q;a\phi + b\psi).$$

For $\phi, \psi : \{0,1\}^n \times Z \to \mathbb{R}$, we write $\phi \succcurlyeq \psi$, if for all $(\sigma, z) \in \{0,1\}^n \times Z$, $\phi(\sigma, z) \geq \psi(\sigma, z)$.

Observation 2. *Let ϕ and ψ be two cost-functions relative to Z. For any decision tree Q over Z and any assignment σ to Z, if $\phi \succcurlyeq \psi$, then*

$$\mathrm{cost}(Q;\phi;\sigma) \geq \mathrm{cost}(Q;\psi;\sigma).$$

Thus, for any distribution μ,

$$\mathrm{cost}_{\mu}(Q;\phi) \geq \mathrm{cost}_{\mu}(Q;\psi).$$

2.3 Definitions pertaining to trees

For a rooted tree T, the *depth* of a leaf is the number of edges on the path to the root. The *depth* of the tree is the maximum depth of a leaf. We denote by L_T the set of its leaves and by V_T the set of its internal nodes. Define the set of *leaf-parents* of T, P_T, as the set of all nodes in V_T all of whose children are leaves. For $S \subseteq P_T$ let $L_T(S)$ be the set of the leaves of the nodes in S. We call a tree *uniform* if all the leaves are on the same level. A tree such that every node has exactly three children is called *ternary*. For a positive integer d, let U_d denote the uniform ternary tree of depth d.

In the following, let T denote a ternary tree with n leaves. We define a distribution μ_T over $\{0,1\}^n$. The distribution is placing positive weight on inputs that we consider intuitively difficult. Let

$$M_0 = \{(0,0,1),(0,1,0),(1,0,0)\} \text{ and } M_1 = \{(0,1,1),(1,0,1),(1,1,0)\}.$$

In the following definition we view T as a circuit with every internal node labeled by a maj-gate. We denote the corresponding function by F_T.

Definition 1 (Difficult inputs, distribution μ_T). *Call an input to a ternary tree* difficult*, if it is such that the inputs to every gate belong to $M_0 \cup M_1$. Let μ_T, the* difficult distribution for T*, be the uniform distribution over all difficult inputs and write $\mu_d \equiv \mu_{U_d}$.*

Suppose the inputs to a gate, under an assignment σ, belong to M_0 (M_1). We call an input to this gate a *minority under σ* if it has the value 1 (0) and a *majority* otherwise.

3 Proof outline

Our goal is to prove a lower bound on the expected cost of any randomized decision tree Q_R that computes maj_d with bounded error δ. We now discuss the outline of our proof. We start with the tree $T \equiv U_d$ that represents maj_d, the natural cost-function ψ that charges 1 for any query, and the difficult distribution $\mu \equiv \mu_T$. We define a process that shrinks tree T to a smaller tree T' and a corresponding randomized decision tree Q'_R that computes $F_{T'}$ with bounded error δ. The crucial part is to show that for a "more expensive" cost-function ψ',

$$\text{cost}_\mu(Q_R;\psi) \geq \text{cost}_{\mu'}(Q'_R;\psi'),$$

where $\mu' \equiv \mu_{T'}$. The quality of our lower bound will depend on how much more expensive ψ' is than ψ.

The main ingredient in this framework is the shrinking process. A natural choice would be to shrink T by removing three leaves u, v, w so that their parent s would become a leaf in T'. Then, if we had a good algorithm Q for F_T we could design an algorithm Q' for $F_{T'}$ as follows. On input σs, Q' would simulate Q on one of the inputs $\sigma 01 s$, $\sigma 10 s$, $\sigma 0 s 1$, $\sigma 1 s 0$, $\sigma s 01$, $\sigma s 10$, with equal probability. We will show in the next section that such a shrinking process can give an alternate—and simpler—proof of the $\Omega\left(2.5^d\right)$ lower bound of Magniez, Nayak, Santha, and Xiao [6].

To improve their bound we are going to shrink nine leaves to three at a time instead of three to one. This is made precise by the following definition.

Definition 2 (shrink$(T; s)$)**.** *For a ternary tree T, let s be the parent of $u, v, w \in P_T$. Define* shrink$(T; s)$ *as the tree with the children of u, v, w removed (so that $u, v, w \in L_{T'}$, where $T' \equiv \mathrm{shrink}(T; s)$).*

After shrinking our initial tree T to a tree T', we need to define a randomized decision tree Q'_R that will compute $F_{T'}$ with error at most δ. We do so, by giving for each deterministic tree Q that Q_R may choose, a randomized tree Q'. This is the object of the following definition.

Definition 3. *Let Q be any deterministic decision tree for F_T. We define a randomized decision tree Q' for $F_{T'}$. The algorithm Q' on input σuvw chooses σ_u, σ_v, σ_w independently and uniformly at random from*

$$\{(x,0,1),(x,1,0),(0,x,1),(1,x,0),(0,1,x),(1,0,x)\},$$

where x is u, v, w respectively. Then, Q' simulates Q on input $\hat{\sigma} = \sigma\sigma_u\sigma_v\sigma_w$.

This induces a randomized algorithm Q'_R for $F_{T'}$ such that a deterministic tree Q^* is chosen by Q'_R with probability $p'(Q^*) = \sum_Q p(Q) \cdot \Pr[Q' \text{ chooses } Q^*]$.

Observation 3. *If Q_R is a δ-error randomized decision tree for F_T, then Q'_R is a δ-error randomized decision tree for $F_{T'}$.*

Proof. The error of Q'_R on an input σ is bounded by the expected (over the choices of σ_u, σ_v, σ_w) error of Q_R on $\hat{\sigma}$. □

It will be useful to express $\mathrm{cost}_{\mu'}(Q'; \psi')$, for some cost-function ψ', in terms of Q. We have the following proposition.

Proposition 4. *For a ternary tree T and s the parent of $u, v, w \in P_T$, let T' denote* shrink$(T; s)$*. Let ψ' be a cost-function on T', such that $\psi'(\sigma; z) = \lambda$ for all $\sigma \in \{0,1\}^{|L_T|}$ and $z \in \{u, v, w\}$. Then*

$$\mathrm{cost}_{\mu'}(Q'; \psi') = \mathrm{cost}_{\mu}(Q; \psi^*),$$

where

$$\psi^*(\sigma; z) = \begin{cases} \psi'(\sigma; z), & \text{if } z \in L_T \setminus L_T(u,v,w); \\ 0.5 \cdot \lambda, & \text{if } z \in L_T(u,v,w) \text{ and is a majority under } \sigma; \\ 0, & \text{if } z \in L_T(u,v,w) \text{ and is a minority under } \sigma. \end{cases} \tag{1}$$

Proof. Observe that by the definition of Q', $\Pr[\hat{\sigma} = \sigma] = \mu(\sigma)$. Furthermore, each σ is encountered 2^3 times over the random choices of Q'. For $(i,j,k) \in [3]^3$ define cost-functions for T as follows.

$$\psi_{(i,j,k)}(\sigma; z) = \begin{cases} 0, & \text{if } z \in L_T(u,v,w) \setminus \{u_i, v_j, w_k\}; \\ \lambda, & \text{if } z \in \{u_i, v_j, w_k\}; \\ \psi'(\sigma; z), & \text{otherwise.} \end{cases}$$

We have

$$\mathrm{cost}_{\mu'}(Q'; \psi') = \sum_{\sigma} \mu(\sigma) \frac{1}{2^3} \sum_{(i,j,k)} \mathrm{cost}(Q; \psi_{(i,j,k)}; \sigma),$$

where i (j,k) ranges over the majorities of u (v,w). The proposition follows because, given a σ and (i,j,k) as stated, $\sum_{(i,j,k)} \frac{1}{2^3} \psi_{(i,j,k)} = \psi^*$. ☐

4 Some useful toy problems and lemmas

We are going to ask a question regarding decision trees over $\{0,1\}^6$. Let μ be the uniform distribution over

$$\big\{(u,v) \;\big|\; \big(u \in M_0 \wedge v \in M_1\big) \vee \big(u \in M_1 \wedge v \in M_0\big)\big\}.$$

We are interested in cost-functions ϕ of the following form.

$$\phi(\sigma; z) = \begin{cases} 1, & \text{if } z \text{ is a minority;} \\ \eta, & \text{otherwise.} \end{cases}$$

We seek the minimum (negative) real value of η for which $\mathrm{cost}_{\mu}(Q; \phi) \geq 0$ for any decision tree Q.

Before we proceed with this toy problem, we list all decision trees[1] over $\{0,1\}^3$ (besides a few uninteresting ones that query a third variable even though the first two are equal).

4.1 Taxonomy of the decision trees for maj_1

In the following table, we consider all possible deterministic decision trees[1] Q for three variables, x, y, z, assuming they are always queried in the order x, y, z. We write "and* z" to denote a conditional read. That is, z is queried only if the value of $\mathrm{maj}(x,y,z)$ cannot be determined from the values of x and y. (Decision trees that read z even if $x = y$ are of no interest, neither for maj_d, nor for the toy problem.)

In the last column we calculate $\sum_{\sigma \in M_0} \mathrm{cost}(Q; \phi; \sigma)$. Because of the symmetries involved we can look up the costs for $\sigma \in M_1$ as well. For example, the cost of the decision tree in row (2a) when $\sigma \in M_1$, is the same as the cost of the decision tree in row (2b) when $\sigma \in M_0$.

[1] We abuse the term "decision tree" in Section 4.1, since we are actually listing algorithms that query bits but do not output anything.

	Decision tree	Cost
(1)	if $x = 0$, stop; if $x = 1$, stop.	$1 + 2\eta$
(2a)	if $x = 0$, stop; if $x = 1$, read y.	$1 + 3\eta$
(2b)	if $x = 0$, read y; if $x = 1$, stop.	$2 + 3\eta$
(3a)	if $x = 0$, stop; if $x = 1$, read y and* z.	$1 + 4\eta$
(3b)	if $x = 0$, read y and* z; if $x = 1$, stop.	$2 + 4\eta$
(4)	if $x = 0$, read y; if $x = 1$, read y.	$2 + 4\eta$
(5a)	if $x = 0$, read y; if $x = 1$, read y and* z.	$2 + 5\eta$
(5b)	if $x = 0$, read y and* z; if $x = 1$, read y.	$2 + 5\eta$
(6)	if $x = 0$, read y and* z; if $x = 1$, read y and* z.	$2 + 6\eta$

What we are going to use from this table is that for $\eta \in [-1/2, 0]$ (we are not interested in other values anyway), the decision tree of row (3a) has the minimum cost when $(x, y, z) \in M_0$, and the tree of row (3b) when $(x, y, z) \in M_1$. Their cost is $1 + 4\eta$.

4.2 A simple proof of the $\Omega(2.5^d)$ lower bound

We illustrate the usefulness of this table by sketching a proof of the $\Omega(2.5^d)$ lower bound of Magniez, Nayak, Santha, and Xiao [6].

We first show that $R_\delta^{\mu_1}(\mathrm{maj}_1) \geq 2.5 \cdot R_\delta^{\mu_0}(\mathrm{maj}_0)$. For any algorithm Q_1 for maj_1, consider the algorithm Q_0 for maj_0 that on input u simulates one of $Q_1(01u)$, $Q_1(10u)$, $Q_1(0u1)$, $Q_1(1u0)$, $Q_1(u01)$, $Q_1(u10)$, with equal probability. If Q_1 is a δ-error algorithm for maj_1, then Q_0 is a δ-error algorithm for maj_0. Let ψ_1 be the cost-function for maj_1 defined by $\psi(\sigma; z) = 1$ for all σ and z. Let ψ_0 be the cost-function for maj_0 defined by $\psi(0; u) = \psi(1; u) = 2.5$. Then, as in the proof of Proposition 4, we can show that

$$\mathrm{cost}_{\mu_1}(Q_1; \psi_1) - \mathrm{cost}_{\mu_0}(Q_0; \psi_0) = \mathrm{cost}_{\mu_1}(Q_1; \phi),$$

where we define ϕ with $\eta = -0.25$. One can now verify by examining the table, that for this value of η, there is no (deterministic) decision tree Q that can achieve $\mathrm{cost}_{\mu_1}(Q; \phi) < 0$. Thus, $\mathrm{cost}_{\mu_1}(Q_1; \psi_1) \geq \mathrm{cost}_{\mu_0}(Q_0; \psi_0)$ and the result follows.

Applying this reasoning repeatedly, by shrinking one node in U_d at a time, one can show that $R_\delta^{\mu_d}(\mathrm{maj}_d) \geq 2.5^d \cdot R_\delta^{\mu_0}(\mathrm{maj}_0)$. Finally, it is not hard to show that you have to read a bit with probability at least $1 - 2\delta$ to be able to guess it with error at most δ, thus $R_\delta^{\mu_0}(\mathrm{maj}_0) \geq (1 - 2\delta)$. Putting these together, $R_\delta^{\mu_d}(\mathrm{maj}_d) \geq (1 - 2\delta) \cdot 2.5^d$.

Remark. Note how in the above argument the value of u makes a difference. In particular, if $u = 0$, then the best decision tree is the one on row

(3a), whereas if $u = 1$, it is the one on row (3b). This can be circumvented if we only want a bound for maj_1 (see next proposition), but it is not easy to do the same for the inductive step.

Proposition 5. $R_\delta^{\mu_1}(\mathrm{maj}_1) \geq \frac{8}{3} \cdot R_\delta^{\mu_0}(\mathrm{maj}_0)$.

Proof. As above, let ψ_1 be the cost-function for maj_1 defined by $\psi(\sigma; z) = 1$ for all σ and z. Let ψ_0 be the cost-function for maj_0 defined by $\psi(0; u) = \psi(1; u) = 8/3$. Then, as in the proof of Proposition 4, we can show that

$$\mathrm{cost}_{\mu_1}(Q_1; \psi_1) - \mathrm{cost}_{\mu_0}(Q_0; \psi_0) = \mathrm{cost}_{\mu_1}(Q_1; \phi),$$

where we define ϕ with $\eta = -1/3$. Observe now that for any deterministic algorithm Q, $\sum_{\sigma \in M_0 \cup M_1} \mathrm{cost}(Q; \phi; \sigma) \geq 0$. The zero is achieved by the tree on row (6) of the table. Thus, $\mathrm{cost}_{\mu_1}(Q_1; \psi_1) \geq \mathrm{cost}_{\mu_0}(Q_0; \psi_0)$ and the result follows. ☐

4.3 Solution of the toy problem and a corollary

We now return to the toy problem (see the beginning of Section 4 for its statement) and show that we can have $\eta = -0.3$. Although it is not stated in the following lemma, it is easily observable from the proof that this value is best possible.

Lemma 6. *For any decision tree Q and $\eta = -0.3$, $\mathrm{cost}_\mu(Q; \phi) \geq 0$.*

Proof. For the proof we are going to do some case analysis, taking advantage of the symmetries involved. Denote the input by (x, y, z, u, v, w), and call (x, y, z) the left side and (u, v, w) the right side. Assume, without loss of generality (due to the symmetry of μ and the fact that we are calculating expected cost), that the variables on the left side are queried in the order x, y, z and on the right side in the order u, v, w. Assume further, that x is the first variable queried by Q, and let Q_0 (Q_1) be the decision tree if $x = 0$ ($x = 1$). Observe that, again due to symmetry, we do not need to analyze Q_1 (the analysis would be the same with the roles of 0 and 1 exchanged). Thus, we assume $x = 0$ and proceed with the analysis of Q_0.

In all of the following cases we calculate the cost scaled; in particular, we calculate $C \equiv \sum_{\sigma : x=0} \mathrm{cost}(Q; \phi; \sigma)$.

Case 1. Suppose that Q_0 is empty. Then $C = 3 + 6\eta > 0$.

Case 2. Suppose that Q_0 queries y. Then, either $x = y$ or $x \neq y$. In the first case, Q_0 should never query z, since such a query is guaranteed to increase the cost by 1. In the second case, Q_0 should definitely query z, since such a query is guaranteed to decrease the cost by $-\eta$. Therefore, we may assume that Q_0 first "finishes" with the left side and then proceeds to the right side, knowing whether $(u, v, w) \in M_0$ or $(u, v, w) \in M_1$. In the first case, Q_0 should continue with the right side as in row (3a) of the table; in the second case, it should continue as in row (3b). The cost is $C = (3 \cdot 2\eta + 1 + 4\eta) + 2 \cdot (3 \cdot (1 + 2\eta) + 1 + 4\eta)$, which is 0 for $\eta = -0.3$.

Case 3. Suppose that Q_0 queries u.

(i) Suppose $x = u$. If Q_0 does not query anything else, then this case contributes to the cost $4 \cdot (1 + \eta)$. Otherwise lets assume (without loss of generality) that it reads y. Then, as in Case 2, we may assume that Q_0 "finishes" the left side before doing anything else. There are four inputs such that $x = u = 0$. For two of the inputs the left side belongs to M_0 and for the other two to M_1. In the first case, Q_0 should read v and w (they are both majorities). In the second case, Q_0 should not read any of v, w (it costs an additional $1 + 2\eta > 0$ if it reads them). In total the cost of this case is then $(1 + 4\eta) + (2 + 4\eta) + 2 \cdot (1 + 3\eta) = 5 + 14\eta$.

(ii) Suppose $x \neq u$. If Q_0 does not query anything else, then this case contributes to the cost $2 + 8\eta$. Otherwise lets assume (without loss of generality) that it reads y. With similar considerations as in case 3(i), we find that the total cost of this case is then $(2 + 4\eta) + 2 \cdot 3\eta + 2 \cdot (1 + 3\eta) = 4 + 16\eta$.

Summing up for case 3, we find that the best Q_0 can do is $C = 9 + 30\eta = 0$. The case analysis is complete. □

We prove a corollary of this lemma that would be more appropriate for our purposes.

Corollary 7. *Let $T \equiv U_2$ with root s and $T' \equiv \mathrm{shrink}(T; s)$. Let ψ and ψ' be cost-functions such that $\psi(\sigma; z) = \lambda \geq 0$ for all $\sigma \in \{0,1\}^9$ and all variables $z \in L_T$, and $\psi'(\sigma; z) = 2.55 \cdot \lambda$ for all $\sigma \in \{0,1\}^3$ and all variables $z \in L_{T'}$. Let μ and μ' be the difficult distributions for $\{0,1\}^9$ and $\{0,1\}^3$ respectively. Then, for any deterministic decision tree Q,*

$$\mathrm{cost}_\mu(Q; \psi) \geq \mathrm{cost}_{\mu'}(Q'; \psi').$$

Proof. Recall the definition of ψ^* from page 6. We have

$$\begin{aligned}
&\mathrm{cost}_\mu(Q; \psi) - \mathrm{cost}_{\mu'}(Q'; \psi') \\
&\quad = \mathrm{cost}_\mu(Q; \psi) - \mathrm{cost}_\mu(Q; \psi^*) && \text{by Proposition 1} \\
&\quad = \mathrm{cost}_\mu(Q; \psi - \psi^*) && \text{by Observation 1} \\
&\quad = \textstyle\sum_{\sigma:\mathrm{maj}_2(\sigma)=0} \mu(\sigma)\, \mathrm{cost}(Q; \psi - \psi^*; \sigma) \\
&\qquad + \textstyle\sum_{\sigma:\mathrm{maj}_2(\sigma)=1} \mu(\sigma)\, \mathrm{cost}(Q; \psi - \psi^*; \sigma).
\end{aligned}$$

We are going to show that the first sum is nonnegative. The other sum can be treated similarly. To that end, we define an intermediate cost-function ξ. In the following definition, σ is an assignment, z a variable, and u is the value of the parent of z under σ.

$$\xi(\sigma; z) = \begin{cases} \lambda, & \text{if } z \text{ is a minority under } \sigma; \\ -0.275 \cdot \lambda, & \text{if } z \text{ is a majority under } \sigma \text{ and } u = 0; \\ -0.3 \cdot \lambda, & \text{if } z \text{ is a majority under } \sigma \text{ and } u = 1. \end{cases}$$

Observe that $\psi - \psi^* \succcurlyeq \xi$ (they agree on the minorities and $\psi - \psi*$ is $\lambda - 0.5 \cdot 2.55 \cdot \lambda = -0.275 \cdot \lambda$ on all majorities) and thus it suffices to show that

$$\textstyle\sum_{\sigma:\mathrm{maj}_2(\sigma)=0} \mu(\sigma)\,\mathrm{cost}(Q;\xi;\sigma) \geq 0. \tag{2}$$

We are now going to decompose ξ into several cost-functions. Let u, v, and w be the children of s. Define a cost-function ξ_u by

$$\xi_u(\sigma;z) = \begin{cases} 0, & \text{if } z \in L_T(u); \\ -0.3 \cdot \lambda, & \text{if } z \text{ is a majority under } \sigma \text{ and } z \in L_T(v,w); \\ \lambda, & \text{if } z \text{ is a minority under } \sigma \text{ and } z \in L_T(v,w). \end{cases}$$

Similarly define ξ_v and ξ_w. For $\alpha \in M_0$ define

$$C_u(\alpha) \equiv \sum_{\beta \in M_0} \sum_{\gamma \in M_1} \mu(\alpha\beta\gamma)\,\mathrm{cost}(Q;\xi_u;\alpha\beta\gamma) + \mu(\alpha\gamma\beta)\,\mathrm{cost}(Q;\xi_u;\alpha\gamma\beta).$$

Similarly define C_v and C_w (assigning α to v and w respectively). Define a cost-function ξ'_u by

$$\xi'_u(\sigma;z) = \begin{cases} 0, & \text{if } z \in L_T(v,w); \\ -0.25 \cdot \lambda, & \text{if } z \text{ is a majority under } \sigma \text{ and } z \in L_T(u); \\ \lambda, & \text{if } z \text{ is a minority under } \sigma \text{ and } z \in L_T(u). \end{cases}$$

Similarly define ξ'_v and ξ'_w. For $(\alpha, \beta) \in M_0 \times M_1$ define

$$C'_u(\alpha,\beta) \equiv \sum_{\gamma \in M_0} \mu(\gamma\alpha\beta)\,\mathrm{cost}(Q;\xi'_u;\gamma\alpha\beta) + \mu(\gamma\beta\alpha)\,\mathrm{cost}(Q;\xi'_u;\gamma\beta\alpha).$$

Similarly define C'_v and C'_w.

We now argue that

$$\begin{aligned}\textstyle\sum_{\sigma:\mathrm{maj}_2(\sigma)=0} \mu(\sigma)\,\mathrm{cost}(Q;\xi;\sigma) = \frac{1}{2}\Bigg[\sum_{\alpha\in M_0}\Big(C_u(\alpha) + C_v(\alpha) + C_w(\alpha)\Big) \\ + \sum_{\alpha \in M_0}\sum_{\beta\in M_1}\Big(C'_u(\alpha,\beta) + C'_v(\alpha,\beta) + C'_w(\alpha,\beta)\Big)\Bigg]. \end{aligned} \tag{3}$$

To prove this, we fix a $\sigma = xyz$ on the left-hand side and see if each bit—assuming it is queried by Q—is charged the same in both sides of the equation. Without loss of generality, lets assume σ is such that $\mathrm{maj}(x) = \mathrm{maj}(y) = 0$ and $\mathrm{maj}(z) = 1$. A minority (under σ) is charged λ on the left side. A minority below u is charged λ once in $C_v(y)$ and once in $C'_u(y,z)$ on the right, for a total of $0.5 \cdot (\lambda + \lambda)$. Similarly for a minority below v. A minority below w will be charged λ in $C_u(x)$ and $C_v(y)$, which corresponds to the amount charged on the left side. A majority below u will be charged $-0.3 \cdot \lambda$ in $C_v(y)$ and $-0.25 \cdot \lambda$ in $C'_u(y,z)$, for a total of $0.5 \cdot (-0.3 - 0.25) \cdot \lambda$; this is how much is charged in the left side as well. Similarly for a majority below v. Finally, a majority below w is charged $-0.3 \cdot \lambda$ in $C_u(x)$ and $C_v(y)$, equal to the amount charged on the left side.

We now finish the proof by showing that the right-hand side of Equation (3) is nonnegative. We argue that, for any $\alpha \in M_0$, $C_u(\alpha) \geq 0$. Each fixed $\alpha \in M_0$ induces a decision tree Q_α over $\{0,1\}^6$ such that $Q_\alpha(\beta\gamma) = Q(\alpha\beta\gamma)$. Observe that ξ_u agrees on $L_T(v,w)$ with $\lambda\phi$. Thus, $C_u(\gamma) = \lambda \operatorname{cost}_\mu(Q_\gamma;\phi)$ and Lemma 6 shows that $C_u(\alpha) \geq 0$. Similarly, for any $\alpha \in M_0$, $C_v(\gamma)$, $C_w(\gamma) \geq 0$. Along similar lines we can show that, for any $(\alpha,\beta) \in M_0 \times M_1$, $C'_u(\alpha,\beta)$, $C'_v(\alpha,\beta)$, $C'_w(\alpha,\beta) \geq 0$. (Lemma 6 is not needed in this case; inspection of the table in Section 4.1 suffices. See also the first paragraph of Section 4.2). $\square$

5 Proof of the lower bound

In this section we carry out the inductive proof sketched in Section 3.

Lemma 8 (Shrinking Lemma). *For a ternary tree T and s the parent of $u,v,w \in P_T$, let T' denote* $\operatorname{shrink}(T;s)$. *Let ψ and ψ' be cost-functions on T and T' such that $\psi(\sigma;z) = \lambda$ for all $\sigma \in \{0,1\}^{|L_T|}$ and $z \in L_T(u,v,w)$ and*

$$\psi'(\sigma;t) = \begin{cases} 2.55 \cdot \lambda, & \text{if } t \in \{u,v,w\}; \\ \psi(\sigma;t), & \text{otherwise.} \end{cases}$$

Then, for any randomized decision tree Q_R,

$$\operatorname{cost}_\mu(Q_R;\psi) \geq \operatorname{cost}_{\mu'}(Q'_R;\psi'),$$

where $\mu \equiv \mu_T$ and $\mu' \equiv \mu_{T'}$.

Proof. Let n denote the number of leaves in T. Fix a partial assignment $\pi \in \{0,1\}^{n-9}$ for the leaves in $L_T \setminus L_T(u,v,w)$ and let $\rho \in \{0,1\}^9$. We write $\pi\rho$ for the assignment that equals ρ on the variables $L_T(u,v,w)$ and π everywhere else. For any deterministic tree Q we have

$$\begin{aligned} \Delta(Q) &\equiv \operatorname{cost}_\mu(Q;\psi) - \operatorname{cost}_{\mu'}(Q';\psi') \\ &= \operatorname{cost}_\mu(Q;\psi) - \operatorname{cost}_\mu(Q;\psi^*) && \text{by Proposition 4} \\ &= \operatorname{cost}_\mu(Q;\psi - \psi^*) && \text{by Observation 1} \\ &= \sum_\pi \sum_\rho \mu(\pi\rho) \operatorname{cost}(Q;\psi - \psi^*;\pi\rho). \end{aligned}$$

Now, ψ and ψ^* are equal over $L_T \setminus L_T(u,v,w)$. Furthermore, having fixed π, we can define a deterministic tree Q_π over $\{0,1\}^9$ so that on input $\rho \in \{0,1\}^9$ we have $Q_\pi(\rho) = Q(\pi\rho)$. Thus

$$\Delta(Q) = \sum_\pi \sum_\rho \mu(\pi\rho) \operatorname{cost}(Q_\pi;\lambda\phi;\rho).$$

This is because, recalling the definition of ψ^* (on page 6), we see that $\psi - \psi^*$ agrees with $\lambda\phi$ on $L_T(u,v,w)$. Thus, we may apply Corollary 7, which implies that, for each fixed π, each summand is greater or equal to zero. It follows that, for any Q, $\Delta(Q) \geq 0$. Finally,

$$\mathrm{cost}_{\mu}(Q_R;\psi) - \mathrm{cost}_{\mu'}(Q'_R;\psi') = \sum_{Q} p(Q)\Delta(Q) \geq 0.$$

$\square$

Applying the Shrinking Lemma repeatedly and recalling Observation 3 it is straightforward to show that

$$R_{\delta}^{\mu_d}(\mathrm{maj}_d) \geq 2.55^{d-1} \cdot R_{\delta}^{\mu_1}(\mathrm{maj}_1).$$

By Proposition 5, $R_{\delta}^{\mu_1}(\mathrm{maj}_1) \geq \frac{8}{3} \cdot R_{\delta}^{\mu_0}(\mathrm{maj}_0)$. A δ-error decision tree for maj_0 should guess a random bit with error at most δ; thus, $R_{\delta}^{\mu_0}(\mathrm{maj}_0) \geq 1 - 2\delta$.

We have obtained the following theorem.

Theorem 9. $R_{\delta}^{\mu_d}(\mathrm{maj}_d) \geq \frac{8}{3} \cdot (1-2\delta) \cdot 2.55^{d-1}$.

6 Concluding remarks

We improved the lower bound on the recursive majority-of-three function using the method of generalized costs of Saks and Wigderson [7]. It seems though that we didn't exploit the full power of this method. A more essential use of generalized costs could, for example, charge differently for reading a majority than for reading a minority. We couldn't implement such an idea.

Our bound is not optimal. We believe that to obtain optimal bounds using a bottom up inductive proof it is not enough to employ a better cost-function. One should find a way to incorporate in the proof the knowledge the algorithm has for all parts of the tree (see the remark on page 8).

Acknowledgements

I thank Mike Saks for useful discussions and corrections in an earlier draft of this paper and Jeff Steif for pointing out a serious error in an earlier version.

References

[1] Harry Buhrman and Ronald de Wolf. Complexity measures and decision tree complexity: a survey. *Theor. Comput. Sci.*, 288(1):21–43, 2002.

[2] Rafi Heiman, Ilan Newman, and Avi Wigderson. On read-once threshold formulae and their randomized decision tree complexity. *Theor. Comput. Sci.*, 107(1):63–76, 1993.

[3] Rafi Heiman and Avi Wigderson. Randomized vs. deterministic decision tree complexity for read-once boolean functions. *Computational Complexity*, 1:311–329, 1991.

[4] T. S. Jayram, Ravi Kumar, and D. Sivakumar. Two applications of information complexity. In *STOC*, pages 673–682. ACM, 2003.

[5] Itamar Landau, Asaf Nachmias, Yuval Peres, and Sithparran Vanniasegaram. The lower bound for evaluating a recursive ternary majority function and an entropy-free proof. *Undergraduate Research Reports, Department of Statistics, University of California, Berkeley*, 2006.

[6] Frédéric Magniez, Ashwin Nayak, Miklos Santha, and David Xiao. Improved bounds for the randomized decision tree complexity of recursive majority. In Luca Aceto, Monika Henzinger, and Jiri Sgall, editors, *ICALP (1)*, volume 6755 of *Lecture Notes in Computer Science*, pages 317–329. Springer, 2011.

[7] Michael E. Saks and Avi Wigderson. Probabilistic boolean decision trees and the complexity of evaluating game trees. In *FOCS*, pages 29–38. IEEE Computer Society, 1986.

[8] Miklos Santha. On the monte carlo boolean decision tree complexity of read-once formulae. *Random Struct. Algorithms*, 6(1):75–88, 1995.

[9] Jonah Sherman. Unpublished, 2007. Communicated to the author by Ryan O'Donnell in 12 Sep. 2012.

[10] Marc Snir. Lower bounds on probabilistic linear decision trees. *Theor. Comput. Sci.*, 38:69–82, 1985.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de