Approximate query complexity

CLIFFORD SMYTH
University of North Carolina Greensboro
Greensboro, NC, USA
cdsmyth@unCG.edu
May 31, 2011

Abstract

Let $f : \{0, 1\}^n \to \{0, 1\}$. Let $\mu$ be a product probability measure on $\{0, 1\}^n$. For $\epsilon \geq 0$ we define $D_\epsilon(f)$, the $\epsilon$-approximate decision tree complexity of $f$, to be the minimum depth of a decision tree $T$ with $\mu(T(x) \neq f(x)) \leq \epsilon$. For $j = 0$ or 1 and for $\delta \geq 0$, we define $C_{j, \delta}(f)$, the $\delta$-approximate $j$-certificate complexity of $f$, to be the minimum certificate complexity of a set $A \subseteq \Omega$ with $\mu(A \Delta f^{-1}(j)) \leq \epsilon$. Note that $D_0(f) = D(f)$ and $C_{j, 0}(f) = C_j(f)$ are the ordinary decision tree and $j$-certificate complexities of $f$, respectively. We extend the well-known result, $D(f) \leq C_1(f) C_0(f)$ [3, 6, 12], proving that for all $\epsilon > 0$ there exists a $\delta > 0$ and a constant $K = K(\epsilon, \delta) > 0$ such that for all $n, \mu, f$, $D_\epsilon(f) \leq K C_{1, \delta}(f) C_{0, \delta}(f)$. We also answer a related question
on query complexity raised by Tardos [12]. We prove generalizations of these results to general product probability spaces.

1 Introduction

Let \( n \geq 1 \) be an integer. For each \( i, 0 \leq i \leq n \), let \( \Omega_i \) be a finite set with at least two elements. Let \( \Omega = \prod_{i=1}^{n} \Omega_i \) and let \( f: \Omega \to \Omega_0 \). Elements \( x \in \Omega \) will be called \textit{assignments} or \textit{inputs} to \( f \) and elements of \( \Omega_0 \) will be called \textit{outputs}. If \( f: \{0, 1\}^n \to \{0, 1\} \), \( f \) is called a \textit{Boolean function}.

Consider a deterministic algorithm \( M \) which \textit{computes} \( f \), i.e. outputs \( M(x) = f(x) \) when given input \( x \). The \textit{query complexity} of \( M \) is the maximum, taken over all inputs \( x = (x_1, \ldots, x_n) \), of the number of coordinates \( x_i \) that \( M \) \textit{queries}, i.e. reads, before returning \( f(x) \). The \textit{deterministic query complexity} of \( f \), written \( D(f) \), is defined to be the minimum query complexity of a deterministic algorithm \( M \) computing \( f \). Note that \( D(f) \leq n \) where \( n \) is the number of input coordinates.

We will analyze our algorithms in the query complexity model: only the number of queries is counted and an algorithm otherwise is assumed to have unlimited computational power. Since only queries are relevant, any such algorithm can be presented in a standardized form known as a decision tree.

A \textit{decision tree} \( T \) with domain \( \Omega \) and range \( \Omega_0 \) is a finite rooted tree such that every internal vertex \( v \) is labeled with a constant \( q(v) \in [n] := \{1, 2, \ldots, n\} \) called the \textit{query coordinate} of \( v \) and each leaf \( l \) is labeled with
a constant $o(l) \in \Omega_0$ called the output of $l$. Each internal vertex $v \in T$ with $q(v) = i$ has $|\Omega_i|$ children, one child $c(v, j)$ for each $j \in \Omega_i$. For each input $x$, we define a path $p_T(x) := (v_0, \ldots, v_d)$ in $T$ as follows. $p_T(x)$ begins with the root $v_0$. For $0 \leq i < d$, $v_{i+1} = c(v_i, x_{q(v_i)})$ and the path terminates upon reaching the leaf $v_d$ that is uniquely determined by $T$ and $x$. We define $T : \Omega \rightarrow \Omega_0$, the function computed by $T$, to be $T(x) := o(v_d)$. We say $T$ computes $f$, if $T(x) = f(x)$ for all $x$.

The query complexity of $T$ on $x$ is the length of $p_T(x)$, the number of queries $T$ makes while computing $T(x)$. $D(T)$, the query complexity of $T$, is the depth of $T$, the maximum length of a path in $T$ from the root to a leaf. Clearly,

$$D(f) = \min \{D(T) : T \text{ is a decision tree computing } f \}.$$ 

Hence $D(f)$ is also known as the decision tree complexity of $f$.

A certificate of $\Omega$ is a function $c$ with domain $\text{dom}(c) \subseteq [n]$ such that for each $i \in \text{dom}(c)$, $c(i) \in \Omega_i$. The complexity or size of $c$ is $|c| := |\text{dom}(c)|$. We say $x$ satisfies $c$, written $x \models c$, if $x_i = c(i)$ for all $i \in \text{dom}(c)$. Let $[c] := \{x \in \Omega : x \models c\}$ be the cylinder of assignments satisfying $c$. The empty certificate, $\lambda$, is the unique certificate with $|\lambda| = 0$.

If $\mathcal{A}$ is a non-empty set of certificates, the certificate complexity of $\mathcal{A}$ is $C(\mathcal{A}) := \max \{|a| : a \in \mathcal{A}\}$. We define $[\mathcal{A}] := \bigcup_{a \in \mathcal{A}} [a]$, the subset of $\Omega$. 

3
represented by \( \mathcal{A} \). If \( A \subseteq \Omega \), we define the certificate complexity of \( A \),

\[
C(A) := \min\{C(\mathcal{A}) : [\mathcal{A}] = A\}.
\]

We adopt the convention that \( C(\mathcal{A}) = 0 \) if \( \mathcal{A} \) is an empty set of certificates and that \( C(A) = 0 \) if \( A = \emptyset \). Note that \( C(\Omega) = 0 \) as \([\lambda] = \Omega\).

Let \( f : \Omega \to \Omega_0 \). If \( j \in \Omega_0 \) and \( x \in f^{-1}(j) \), we say that \( c \) is a \( j \)-certificate for \( x \) with respect to \( f \) if \( x \in [c] \subseteq f^{-1}(j) \). The \( j \)-certificate complexity of \( f \) is \( N_j(f) := C(f^{-1}(j)) \). We may think of \( N_j(f) \) as the non-deterministic query complexity of the finite decision problem: “Given \( x \in \Omega \), is \( f(x) = j \)?”.

If \( x \in f^{-1}(j) \), at most \( N_j(f) \) coordinates of \( x \) need to be queried to prove this.

The following was proven independently in [3], [6], and [12].

**Theorem 1.1**

If \( f : \{0, 1\}^n \to \{0, 1\} \) then

\[
D(f) \leq N_1(f)N_0(f).
\]  \hspace{1cm} (1)

For \( i \in [n] \), let \( \mu_i \) be a discrete probability measure on \( \Omega_i \). Let \((\Omega, \mu)\) be the corresponding finite product probability space, with product measure \( \mu = \prod_{i=1}^{n} \mu_i \). Without loss of generality, we may assume that for all \( i \) and for all \( \omega \in \Omega_i \), \( \mu_i(\omega) > 0 \). Given \( \epsilon \geq 0 \), we define the following natural \( \epsilon \)-approximate versions of decision tree and certificate complexity.
For $A \subseteq \Omega$, we define

$$C_\epsilon(A) := \min \{ C(A) : \mu([A] \Delta A) \leq \epsilon \}$$

where $A \Delta B := (A \setminus B) \cup (B \setminus A)$. Note that $C_0(A) = C(A)$.

For $f : \Omega \rightarrow \Omega_0$ and a decision tree $T$, we say $T$ $\epsilon$-approximates $f$ if $\mu(\{x \in \Omega : T(x) \neq f(x)\}) \leq \epsilon$. We define the $\epsilon$-approximate decision tree complexity of $f$,

$$D_\epsilon(f) := \min \{ D(T) : T \text{ is a decision tree that } \epsilon\text{-approximates } f \}.$$  

If $j \in \Omega_0$ then the $\epsilon$-approximate $j$-certificate complexity of $f$ is defined to be

$$N_{j,\epsilon}(f) := C_\epsilon(f^{-1}(j)).$$

Note that $D_0(f) = D(f)$ and $N_{j,0}(f) = N_j(f)$.

Our main result is the following generalization\(^1\) of Theorem 1.1 to the $\epsilon$-approximate setting.

**Theorem 1.2**

Given $\epsilon > 0$, let $\beta_0 = \epsilon - \epsilon_0$ where $\epsilon_0$ is the unique solution of $\epsilon_0 + (\epsilon_0/3)^3 = \epsilon$.

Then for all $f : \Omega \rightarrow \{0, 1\}$, with $(\Omega, \mu)$ a finite product probability space and

\(^1\)A preliminary and weaker version of this result appeared in [11].
for all $\alpha_1, \alpha_0 \geq 0$, if $\alpha = \alpha_1 + \alpha_0 < \beta_0$ then

$$D_\epsilon(f) \leq K \ N_{1,\alpha_1}(f) \ N_{0,\alpha_0}(f), \quad (2)$$

where $K = (1 - \gamma/3)/((\gamma/3)^3 - \alpha)$ and $\gamma = \epsilon - \alpha$.

Note that $K$ is independent of both the product probability space $(\Omega, \mu)$ and the function $f$.

Let $\mathcal{A}_1, \ldots, \mathcal{A}_m$ be families of certificates. We define

$$U(\mathcal{A}_1, \ldots, \mathcal{A}_m) := \{x \in \Omega : \exists! j \in [m] \ x \in [A_j]\},$$

the set of uniquely covered assignments and

$$\text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m) := \Omega \setminus U(\mathcal{A}_1, \ldots, \mathcal{A}_m),$$

the set of ambiguous assignments. We define

$$\delta(\mathcal{A}_1, \ldots, \mathcal{A}_m) := \mu(\bigcup_{1 \leq i < j \leq m} [A_i] \cap [A_j]).$$

In [12], Tardos defined the $\epsilon$-approximate compatibility game on $\mathcal{A}_1, \ldots, \mathcal{A}_m$ as follows.

**Definition 1.3** The output of a decision tree $T : \Omega \rightarrow \{\mathcal{A}_1, \ldots, \mathcal{A}_m\}$ on $x \in U(\mathcal{A}_1, \ldots, \mathcal{A}_m)$ is defined to be correct if $x \in [T(x)]$ and an error otherwise. The output of $T$ on any $x \in \text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m)$ is defined to be correct. We
say $T$ plays the $\epsilon$-approximate compatibility game on $A_1, \ldots, A_m$, if

$$\mu(\{x \in \Omega : T \text{ makes an error on } x\}) \leq \epsilon.$$ 

Let $G_\epsilon(A_1, \ldots, A_m)$ be the minimum query complexity of such a decision tree. (Tardos’s definition of the $\epsilon$-compatibility game was for the main case, $m = 2$.)

The proofs of Theorem 1.1 given in [3], [6], and [12] are all algorithmic, in effect proving:

**Theorem 1.4**

*If $\delta(A, B) = 0$, then $G_0(A, B) \leq C(A)C(B)$.***

It is easy to see that Theorem 1.1 follows. Let $A, B$ be sets of certificates with $[A] = f^{-1}(1)$, $C(A) = N_1(f)$, $[B] = f^{-1}(0)$, and $C(B) = N_0(f)$. By Theorem 1.4 there exists a decision tree $T$ playing the $\epsilon$-compatibility game on $A$ and $B$ with $D(T) \leq C(A)C(B)$. Since $U(A, B) = \Omega$, $T(x) = f(x)$ for all $x$ and $D(f) \leq N_1(f)N_0(f)$.

Similarly Theorem 1.2 will follow from

**Theorem 1.5**

*Given $\epsilon > 0$ let $\delta_0 = (\epsilon/3)^3$. Then for all finite product probability spaces $(\Omega, \mu)$ and all sets of certificates $A, B$ on $\Omega$, if $\delta = \delta(A, B) < \delta_0$ then

$$G_\epsilon(A, B) \leq K C(A)C(B).$$

(3)
where $K = (1 - \epsilon/3)/((\epsilon/3)^3 - \delta)$.

Tardos made the following conjecture in [12]:

**Conjecture 1.6**

There exists a positive constant $c$ so that the following holds. For all $\epsilon > 0$ there is a $\delta > 0$ such that for any $n \geq 1$ and any sets of certificates $\mathcal{A}, \mathcal{B}$ on \{0, 1\}$^n$ with $|\text{amb}(\mathcal{A}, \mathcal{B})| \leq \delta 2^n$, $G_\epsilon(\mathcal{A}, \mathcal{B}) \leq m^\epsilon$ where $m = \max\{C(\mathcal{A}), C(\mathcal{B})\}$.

Theorem 1.5 may be viewed as only a partial answer to this conjecture; when $\epsilon \to 0$ the constant $K$ of Theorem 1.5 goes to infinity.

We define certificates $a$ and $b$ to be disjoint, denoted $a \sim^d b$ if $\text{dom}(a) \cap \text{dom}(b) = \emptyset$. If $A, B \subseteq \Omega$ let

$$A \times^d B := \{(x, y) : \exists \text{ certificates } a \sim^d b \; x \in [a] \subseteq A, y \in [b] \subseteq B\}.$$  

The crux of the proof Theorem 1.5 is

**Theorem 1.7 (The Dual Inequality [8])** Let $(\Omega, \mu)$ be a finite product probability space and $A, B \subseteq \Omega$. Then

$$(\mu \times \mu)(A \times^d B) \leq \mu(A \cap B).$$

We will use the following version of Theorem 1.7 applied to sets of certificates $\mathcal{A}$ and $\mathcal{B}$,

$$(\mu \times \mu) \left( \bigcup_{a \in \mathcal{A}, b \in \mathcal{B}; a \sim^d b} [a] \times [b] \right) \leq \mu([\mathcal{A}] \cap [\mathcal{B}]).$$  

(4)
The dual inequality is so named due to its similarity to Reimer’s inequality (also known as the BKR inequality). If $A, B \subseteq \Omega$ let

$$A \cap^d B := \{ x : \exists \text{ certificates } a \sim^d b \ x \in [a] \subseteq A, x \in [b] \subseteq B \}. $$

**Theorem 1.8 (Reimer’s Inequality [10])** Let $(\Omega, \mu)$ be a finite product probability space and $A, B \subseteq \Omega$. Then

$$\mu(A \cap^d B) \leq \mu(A)\mu(B) = (\mu \times \mu)(A \times B).$$

Theorem 1.8 was conjectured in [2] and proved in [10]. Results extending Theorems 1.7 and 1.8 are given in [4].

We prove Theorems 1.2 and 1.5 in section 2. In the same section we show that Theorem 2.1, a generalization of Theorem 1.5 to the case in which there are $m > 2$ families of certificates, follows immediately as does Theorem 2.2, a generalization of Theorem 1.2 to functions taking on $m > 2$ values. In the final section we note some possible directions for future work.

## 2 Proofs

If $a, b$ are certificates, we say they are *compatible*, denoted $a \sim^c b$, if $a(i) = b(i)$ for all $i \in \text{dom}(a) \cap \text{dom}(b)$. If $a$ and $b$ are compatible let $a \setminus b$ be the certificate with $\text{dom}(a \setminus b) := \text{dom}(a) \setminus \text{dom}(b)$ and $(a \setminus b)(i) := a(i)$ for all $i \in \text{dom}(a \setminus b)$. 
The following proof of Theorem 1.4 (along the lines of [3], [6], and [12]) will form the basis of our proof of Theorem 1.5.

**Proof of Theorem 1.4.** The algorithm $T$ proceeds in rounds. For each round $i \geq 0$, let $q_i = q_i(x)$ be the certificate recording all the queries asked by $T$ up to the beginning of round $i$ when running on input $x$. Let $\text{dom}(q_i)$ be the set of queried variables and let $q_i(j) = x_j$ for each $j \in \text{dom}(q_i)$. Let $Q_i = Q_i(x) := [q_i]$. Let $A_i = A_i(x) := \{ a \setminus q_i : a \in A, a \sim^c q_i \}$ be the set of reduced certificates of $A$ compatible with $q_i$. We define $B_i = B_i(x)$, the reduced certificates of $B$, analogously. Note that if $a \sim^c q_i$, then $[a \setminus q_i] \cap [q_i] = [a] \cap [q_i]$ and so $[A_i] \cap Q_i = [A] \cap Q_i$ and $[B_i] \cap Q_i = [B] \cap Q_i$. We fix a total ordering of the certificates in $A$ and in $B$ and give $A_i$ and $B_i$ the respective induced orderings.

**Algorithm for the $\epsilon$-compatibility game, $\epsilon = 0$.**

Set $i = 0, q_0 = \lambda, Q_0 = \Omega, A_0 = A$ and $B_0 = B$. Repeatedly carry out the following steps until the algorithm halts.

Begin round $i$:

If $B_i = \emptyset$, halt and output $T(x) = A$.

If $A_i = \emptyset$, halt and output $T(x) = B$.

Select the first certificate $a \in A_i$ and query $x_j$ for each $j \in \text{dom}(a)$.

Calculate $q_{i+1}, Q_{i+1} = [q_{i+1}], A_{i+1}$ and $B_{i+1}$ accordingly.

End round $i$ and increment $i$ to $i + 1$. 
If \( T \) terminates on \( x \), its output is correct according to the \( \epsilon \)-compatibility game, see Definition 1.3. We now show that \( T \) always terminates in at most \( C(\mathcal{A})C(\mathcal{B}) \) queries. Suppose that in round \( i \) neither halting condition holds. Let \( a_i \) be the first certificate in \( \mathcal{A}_i \). Since \( [\mathcal{A}] \cap [\mathcal{B}] = \emptyset \), we have \( [\mathcal{A}_i] \cap [\mathcal{B}_i] \cap Q_i = [\mathcal{A}] \cap [\mathcal{B}] \cap Q_i = \emptyset \). Since the conditions \( a \sim^c b \) and \([a] \cap [b] \neq \emptyset\) are equivalent, every \( b \in \mathcal{B}_i \) must be incompatible with every \( a \in \mathcal{A}_i \). That is, we must have \( b(j) \neq a(j) \) for some \( j \not\in \text{dom}(q_i) \). Thus querying \( x_j \) for all \( j \in \text{dom}(a_i) \) yields \( C(\mathcal{B}_{i+1}) \leq C(\mathcal{B}_i) - 1 \). Thus we have \( C(\mathcal{B}_i) = 0 \) at the beginning of some round \( i \leq C(\mathcal{B}) \), i.e. we have \( \mathcal{B}_i = \emptyset \) or \( \mathcal{B}_i = \{\lambda\} \). In the second case, \( \mathcal{A}_i = \emptyset \) as no certificate can be incompatible with \( \lambda \). Therefore the algorithm halts after at most \( C(\mathcal{B}) \) rounds of at most \( C(\mathcal{A}) \) queries each.

\[\begin{align*}
\text{If } x \in [\mathcal{A}] \text{ we define} \\
C(\mathcal{A}, x) & := \min\{|a| : x \models a \in \mathcal{A}\}. \\
\text{Using our total ordering on } \mathcal{A}, \text{ we define} \\
\min(\mathcal{A}, x) & := \min\{a \in \mathcal{A} : x \models a, |a| = C(\mathcal{A}, x)\}.
\end{align*}\]

For the definition of \( \min(\mathcal{A}, x) \) recall that \( \mathcal{A} \) is totally ordered and hence a minimum element may be selected from any non-empty subset. For \( a \in \mathcal{A} \)
we define $[a]' := \{ x \in [a] : \min(\mathcal{A}, x) = a \}$. Note that $\{ [a]' : a \in \mathcal{A} \}$ is a partition of $[\mathcal{A}]$. We define $\min(\mathcal{B}, x)$ and $[b]'$ for $b \in \mathcal{B}$ similarly. Let $q_i, Q_i, \mathcal{A}_i, \mathcal{B}_i$ be as in the proof of Theorem 1.4.

**Proof of Theorem 1.5.** Let $\mathcal{A}, \mathcal{B}, \epsilon$ be given. Let $\delta = \delta(\mathcal{A}, \mathcal{B}) = \mu([\mathcal{A}] \cap [\mathcal{B}])$. Pick $\epsilon_1, \epsilon_2, \epsilon_3 > 0$ so that $\epsilon_1 + (1 - \epsilon_1)(\epsilon_2 + \epsilon_3) \leq \epsilon$. Let $K' = \epsilon_2(1 - \epsilon_1)/(\epsilon_1^2 \epsilon_2 - \delta)$. If $\delta < \epsilon_1^2 \epsilon_2$, we define a decision tree $T$, below, that plays the $\epsilon$-compatibility game on $\mathcal{A}$ and $\mathcal{B}$ with $D(T) \leq (1/\epsilon_3)K' C(\mathcal{A})C(\mathcal{B})$. Taking $\epsilon_1 = \epsilon_2 = \epsilon_3 = \epsilon/3$ gives the statement of the theorem, with $K = (1/\epsilon_3)K' = (1 - \epsilon/3)/((\epsilon/3)^3 - \delta)$.

**Algorithm for the $\epsilon$-compatibility game, $\epsilon > 0$.**

Set $i = 0$, $q_0 = \lambda$, $Q_0 = \Omega$, $\mathcal{A}_0 = \mathcal{A}$ and $\mathcal{B}_0 = \mathcal{B}$. Repeatedly carry out the following steps until the algorithm halts.

Begin round $i$:

Test halting conditions 1:

If $\mu([\mathcal{A}_i]|Q_i) > 1 - \epsilon_1$ or $\mu([\mathcal{B}_i]|Q_i) < \epsilon_1$, halt and output $\mathcal{A}$.

If $\mu([\mathcal{B}_i]|Q_i) > 1 - \epsilon_1$ or $\mu([\mathcal{A}_i]|Q_i) < \epsilon_1$, halt and output $\mathcal{B}$.

Test halting condition 2:

If $\mu([\mathcal{A}_i] \cap [\mathcal{B}_i]|Q_i) > (1/\epsilon_2)\delta$, halt and output either $\mathcal{A}$ or $\mathcal{B}$.

Test halting Condition 3:

If $i > (1/\epsilon_3)K'C(\mathcal{B})$, halt and output either $\mathcal{A}$ or $\mathcal{B}$. 

12
Update: Select the first certificate $a \in A_i$ such that

$$
\mu \left( \bigcup_{b \in B_i : b \neq a} [b] \bigg| Q_i \right) \geq 1/K'
$$

and query all of its variables. Calculate $q_{i+1}, Q_{i+1} = [q_{i+1}], A_{i+1}$ and $B_{i+1}$ accordingly. End round $i$ and increment $i$ to $i + 1$.

The output in halting condition 2 or 3 can be arbitrary without affecting the analysis of the algorithm. We will soon show how the term satisfying (5) can be found in each update step.

We only consider the nodes $v$ of $T$ that correspond to the beginning of some round $i$. Let $r(v) = i$ be the round number of such a vertex $v$. Given $v$, let $q(v), Q(v), A(v), B(v)$, be the status of the variables $q_i, Q_i, A_i, B_i$, respectively, at the beginning of the round corresponding to $v$.

We define a sequence $\{T_j\}_{j=0}^m$ of labeled subtrees of $T$. Vertices of $T_j$ are either labeled unexpanded or expanded. Let $T_0$ be the tree consisting only of the root $u$ of $T$ where $u$ is labeled as unexpanded. For all $j \geq 0$, $T_{j+1}$ is created from $T_j$ as follows. Pick $u_j$, an unexpanded leaf of $T_j$. Let $\tilde{q} = q(u_j), \tilde{Q} = Q(u_j), \tilde{A} = A(u_j), \text{ and } \tilde{B} = B(u_j)$. Let $T_{j+1}$ be the tree that results from $T_j$ by labeling $u_j$ as expanded and running one round of the algorithm on the inputs in $\tilde{Q}$ alone. If halting condition $h$ applies, give $u_j$ an additional halting label $h$. Otherwise, let $a_j \in \tilde{A}$ be the certificate found in the updating step and add a new unexpanded child to $u_j$, one for every outcome of querying the variables in $a_j$. The process ends with $T_m$, a copy
of $T$ in which all vertices are expanded and all leaves have halting labels.

We now show that the certificate $a_j \in \widetilde{A}$ can be found in the update step. If $\mu(\tilde{Q}) > 0$, we define the conditional probability measure, $\mu(X|\tilde{Q}) := \mu(X \cap \tilde{Q})/\mu(\tilde{Q})$ for all $X \subseteq \Omega$. This is a product measure; for all $y \in \Omega$, $\mu(y|\tilde{Q}) = \nu(y) := \prod_{k=1}^{n} \nu_k(y_k)$ where $\nu_k = \mu_k$ for all $k \in [n] \setminus \text{dom}(\tilde{q})$ and $\nu_k = \delta_{x_k}$ for $k \in \text{dom}(\tilde{q})$. Here $\delta_z$ is the probability measure on $\Omega_k$ with $\delta_z(z) = 1$.

By halting conditions 1 and 2 on $Q$, we have $\epsilon_1 \leq \nu([\tilde{A}]), \nu([\tilde{B}]) \leq 1 - \epsilon_1$ and $\nu([\tilde{A}] \cap [\tilde{B}]) \leq (1/\epsilon_2)\delta$. Applying the dual inequality, (4), to $\tilde{A}$ and $\tilde{B}$ on $(\Omega, \nu)$ gives

$$(\nu \times \nu)(\bigcup_{a \sim b} [a]' \times [b]') \leq (\nu \times \nu)(\bigcup_{a \sim b} [a] \times [b]) \leq \nu([\tilde{A}] \cap [\tilde{B}]) \leq (1/\epsilon_2)\delta.$$ 

Here and in what follows, the indices $a$ and $b$ range over $\tilde{A}$ and $\tilde{B}$ respectively. We also have

$$(\nu \times \nu)(\bigcup_{a \neq b} [a]' \times [b]') + (\nu \times \nu)(\bigcup_{a \sim b} [a]' \times [b]') = (\nu \times \nu)(\bigcup_{a,b} [a]' \times [b]')$$

$$= (\nu \times \nu)([\tilde{A}] \times [\tilde{B}]) = \nu([\tilde{A}])\nu([\tilde{B}]) \geq \epsilon_1^2$$

These inequalities give

$$\sum_a \nu([a]')\nu(\bigcup_{b \neq a} [b]') = (\nu \times \nu)(\bigcup_{a \neq b} [a]' \times [b]') \geq \epsilon_1^2 - (1/\epsilon_2)\delta.$$
Since $\lambda := \sum_a \nu([a']) = \nu([\tilde{A}]) \leq 1 - \epsilon_1$, we can divide this last inequality by $\lambda$ to get

$$\sum_a w_a \nu(\bigcup_{b:b \not\sim_d a} [b']) \geq 1/K'$$

where $w_a = \nu([a'])/\lambda \geq 0$, $\sum_a w_a = 1$, and $K' = \epsilon_2(1 - \epsilon_1)/(\epsilon_1^2 \epsilon_2 - \delta)$. Thus by convexity there is a term $a \in \tilde{A}$ satisfying (5), $\mu(\bigcup_{b:b \not\sim_d a} [b']|\tilde{Q}) \geq 1/K'$.

Given $x \in \Omega$, $0 \leq j \leq m$ let $v_j(x)$ be the leaf of $T_j$ reached when $T$ runs on $x$. Let $r_j(x)$ be the number of rounds taken to reach $v_j(x)$. Let $q_j(x), Q_j(x), A_j(x), B_j(x)$ be the status of the the variables $q, Q, A, B$ at $v_j(x)$. Let

$$\ell_j(x) = \begin{cases} 
C(B_j(x), x) & \text{if } x \in [B_j(x)] \\
0 & \text{otherwise.}
\end{cases}$$

Suppose $v_j(x) = u_j$. Let $w = v_{j+1}(x)$. If $\ell_j(x) > 0$, then $\ell_j(x) = |b|$ where $b = \min(B(x), x)$. Since $x \models b$ we have $x \in [b] \cap Q_{j+1}(x) \neq \emptyset$, and thus $b$ is not removed from $B_j(x)$ in the updating step. A copy of $b$, perhaps with some variables deleted, remains in $B_{j+1}(x)$. Thus if $a_j \not\sim_d b$, we have $\ell_{j+1}(x) \leq \ell_j(x) - 1$. But by the choice of $a_j$, the fraction of $x \in \tilde{Q}$ that satisfy this condition is large, $\mu(\bigcup_{b:b \not\sim_d a_j} [b']|\tilde{Q}) \geq 1/K'$, and we have

$$E[\ell_{j+1}(x)|\tilde{Q}] \leq E[\ell_j(x)|\tilde{Q}] - 1/K'$, \tag{6}$$

where the expectation is taken with respect to $\mu$.

Let $E = \{x \in U(\mathcal{A}, \mathcal{B}) : T \text{ makes an error on } x\}$. We show $\mu(E) < \epsilon$. 

15
For \(1 \leq h \leq 3\) let

\[ H_h := \{ x \in \Omega : T \text{ halts on } x \text{ due to halting condition } h \}, \quad E_h := E \cap H_h \]

We will show \(\mu(E_1) < \epsilon_1 \mu(H_1), \mu(H_2) < \epsilon_2,\) and \(\mu(H_3) \leq \epsilon_3\) so that

\[ \mu(E) < \epsilon_1 \mu(H_1) + \mu(H_2) + \mu(H_3) \leq \epsilon_1 + (1 - \epsilon_1) (\epsilon_2 + \epsilon_3) \leq \epsilon \]

where the last inequality holds by the choice of the \(\epsilon_i\). Note that \(T\)'s output when halting conditions 2 or 3 apply is irrelevant to this analysis.

For any \(T_j, \{Q_j(v) : v \text{ a leaf of } T\}\) is a partition of \(\Omega\). We have \(\mu(E_1) = \sum_Q \mu(E_1|Q(v))\mu(Q(v))\) where the sum is over all leaves \(v\) of \(T_m = T\) with halting label 1. For these leaves we have \(\mu(E_1|Q) < \epsilon_1\) and thus \(\mu(E_1) < \epsilon_1 \mu(H_1)\). (If \(\mu([A_i]|Q) > 1 - \epsilon_1\), output due to halting condition 1 is \(A\) and this will be in error iff \(x \in [B_i] \setminus [A_i]\), but \(\mu([B_i] \setminus [A_i]|Q) \leq \epsilon_1\). The other cases follow similarly.)

Clearly \(\delta = \mu([A] \cap [B]) \geq \sum_v \mu([A(v)] \cap [B(v)]|Q(v))\mu(Q(v))\) where the sum is over leaves \(v\) with halting label 2. Since \(\mu([A] \cap [B]|Q) > (1/\epsilon_2)\delta\) for such \(v\), \(\mu(H_2) < \epsilon_2\). It remains to show \(\mu(H_3) \leq \epsilon_3\).

We return to the step passing from \(T_j\) to \(T_{j+1}\). Since we have \(\ell_{j+1}(x) \leq \ell_j(x)\) for all \(x \notin \tilde{Q}\), we may use (6) to get

\[ \mathbb{E}[\ell_j(x)] = \sum_{Q \neq \tilde{Q}} \mathbb{E}[\ell_j(x)|Q] \mu(Q) + \mathbb{E}[\ell_j(x)|\tilde{Q}] \mu(\tilde{Q}) \]
\[ \geq \sum_{Q \neq \tilde{Q}} \mathbb{E}[\ell_{j+1}(x)|Q] \mu(Q) + \mathbb{E}[\ell_{j+1}(x)|\tilde{Q}] \mu(\tilde{Q}) + (1/K') \mu(\tilde{Q}) \\
= \mathbb{E}\ell_{j+1}(x) + (1/K') \mu(Q(v)) \]

where the sums are over all \( Q = Q(v) \) where \( v \) is a leaf of \( T_j \). Thus we have

\[ \mathbb{E}\ell_j(x) - \mathbb{E}\ell_{j+1}(x) \geq (1/K')\mu(\tilde{Q}). \]

Since \( r_{j+1}(x) = r_j(x) + 1 \) for \( x \in \tilde{Q} \) and \( r_{j+1}(x) = r_j(x) \) otherwise, we have \( \mathbb{E}r_{j+1}(x) - \mathbb{E}r_j(x) = \mu(\tilde{Q}) \). Thus

\[ \mathbb{E}\ell_j(x) - \mathbb{E}\ell_{j+1}(x) \geq (1/K')(\mathbb{E}r_{j+1}(x) - \mathbb{E}r_j(x)). \]

Summing over \( 0 \leq j \leq m - 1 \) gives

\[ \mathbb{E}\ell_0(x) - \mathbb{E}\ell_m(x) \geq (1/K')(\mathbb{E}r_m(x) - \mathbb{E}r_0(x)). \]

Since \( r_0(x) = 0 \) and \( \ell_0(x) \leq C(B) \),

\[ K'C(B) \geq \mathbb{E}r_m(x). \]

By Markov’s inequality, \( \mu(H_3) \leq \mu(\{x : r_m(x) > (1/\epsilon_3)K'C(B)\}) \leq \mu(\{x : r_m(x) \geq (1/\epsilon_3)\mathbb{E}r_m(x)\}) \leq \epsilon_3. \)

By halting condition 3, \( T \) performs \( \leq (1/\epsilon_3)K'C(B) \) rounds, each requiring at most \( C(A) \) queries. Thus \( D(T) \leq (1/\epsilon_3)K'C(A)C(B) \) as claimed.
Theorem 1.5 can be generalized to three or more families of certificates.

**Theorem 2.1**

Given $\epsilon > 0$ and $m \geq 2$, let $\delta_0 = (\epsilon/(3b))^3$ where $b = \lceil \log_2 m \rceil$. Then for all finite product probability spaces $(\Omega, \mu)$ and all sets of certificates $A_0, \ldots, A_{m-1}$ on $\Omega$, if $\delta = \delta(A_0, \ldots, A_{m-1}) < \delta_0$ then

$$G_\epsilon(A_0, \ldots, A_{m-1}) \leq bK M_1 M_2$$

(7)

where $K = (1 - \epsilon/(3b))/((\epsilon/(3b))^3 - \delta)$ and where $M_1$ and $M_2$ are the largest and second largest members of $\{C(A_j) : 0 \leq j < m\}$, respectively.

**Proof:** The case $m = 2$ is Theorem 1.5. Suppose $m \geq 3$ and let $b = \lceil \log_2 m \rceil$. For $0 \leq i \leq b - 1$ and for $j = 0$ or $1$ let $S_{i,j}$ be the set of all integers $k \in \{0, \ldots, m - 1\}$ whose $i$th bit in binary is $(k)_i = j$. Let $B_{i,j} = \bigcup_{k \in S_{i,j}} A_k$. Since $\delta(B_{i,0}, B_{i,1}) \leq \delta(A_1, \ldots, A_m) = \delta < \delta_0 = (\epsilon/(3b))^3$, Theorem 1.5 implies we can construct a decision tree $T_i$ with $D(T_i) \leq KM_1 M_2$, playing the $(\epsilon/b)$-compatibility game on $B_{i,1}$ and $B_{i,0}$. Let $T$ be the decision tree formed by running each $T_i$ and giving output $T(x) = A_k$ if and only if $T_i(x) = B_{i,(k)_i}$ for all $0 \leq i \leq b - 1$.

Let $E, E_0, \ldots, E_{b-1}$ be the error sets of $T, T_0, \ldots, T_{b-1}$ respectively. To prove the theorem, it suffices to show $E \subseteq \bigcup_{i=0}^{b-1} E_i$. If $x \in E$, then $x \in U(A_0, \ldots, A_{m-1})$ and there exists $k \neq k'$ such that $x \in [A_k]$ while $T(x) =$
A′. Let $0 \leq i \leq b - 1$ such that $(k)_i \neq (k')_i$. If $(k)_i = 1$, $x \in [B_{i,1}] \setminus [B_{i,0}]$ and $T_i(x) = B_{i,0}$, otherwise $x \in [B_{i,0}] \setminus [B_{i,1}]$ and $T_i(x) = B_{i,1}$. In either case, $x \in E_i$.

Theorem 1.2 is the case $m = 2$ of the following

**Theorem 2.2**

*Given* $\epsilon > 0$ and $m \geq 2$, let $\beta_0 = \epsilon - \epsilon_0$ where $\epsilon_0$ is the unique solution of $\epsilon_0 + (\epsilon_0/(3b))^3 = \epsilon$ and $b = \lceil \log_2 m \rceil$. Then for all $f : \Omega \to [m]$, with $(\Omega, \mu)$ a finite product probability space and for all $\alpha_1, \ldots, \alpha_m \geq 0$, if $\alpha = \sum \alpha_j < \beta_0$ then

$$D_{\epsilon}(f) \leq bK M_1 M_2$$

(8)

where $K = (1 - \gamma/(3b))/((\gamma/(3b))^3 - \alpha)$, $\gamma = \epsilon - \alpha$, and where $M_1$ and $M_2$ are the largest and second largest values of $\{N_{j,\alpha_j}(f) : j \in \Omega_0\}$, respectively.

Note the constant $K$ in Theorem 2.2 is independent of $(\Omega, \mu)$ and $f$.

**Proof:** Let $\mathcal{A}_1, \ldots, \mathcal{A}_m$ be sets of certificates with $C(A_i) = N_{i,\alpha_i}(f)$ and $\mu(E_{A_i}) \leq \alpha_i$ for all $i \in [m]$ where $E_{A_i} = [A_i]\Delta f^{-1}(i)$. Let $\alpha = \sum \alpha_i$. We claim $\text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m) \subseteq \bigcup_{i[m]} E_{A_i}$. Note that if $x \in \text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m)$ and $f(x) = i$ then either $x \notin [A_i]$ and so $x \in E_{A_i}$ or $x \in [A_i] \cap [A_j]$ for some $j \neq i$ and so $x \in E_{A_j}$. Thus $\delta(\mathcal{A}_1, \ldots, \mathcal{A}_m) \leq \mu(\text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m)) \leq \alpha$.

By Theorem 2.1, if $\alpha < (\gamma/(3b))^3$ there is a decision tree $T$ that plays the
\( \gamma \)-approximate compatibility game on \( \mathcal{A}_1, \ldots, \mathcal{A}_m \) with query complexity

\[
G_\gamma(\mathcal{A}_1, \ldots, \mathcal{A}_m) \leq bK M_1 M_2 \tag{9}
\]

where \( K = (1 - \gamma/(3b))/((\gamma/(3b))^3 - \alpha) \). We change the definition of \( T \) slightly so that \( T \) maps \( \Omega \) to \([m]\) instead of \( \{\mathcal{A}_1, \ldots, \mathcal{A}_m\} \) and \( T(x) = \mathcal{A}_i \) is instead represented as \( T(x) = i \).

Let \( E_U = \{x \in U(\mathcal{A}_1, \ldots, \mathcal{A}_m) : T \text{ makes an error on } x\} \). By construction of \( T \), \( \mu(E_U) \leq \gamma \). It can be seen by a straightforward checking of cases, that

\[
\{x \in \Omega : T(x) \neq f(x)\} \subseteq E_U \cup \bigcup_{i \in [m]} E_{\mathcal{A}_i}.
\]

Since \( \text{amb}(\mathcal{A}_1, \ldots, \mathcal{A}_m) \subseteq \bigcup_{i \in [m]} E_{\mathcal{A}_i} \), it suffices to show the preceding inclusion for all \( x \in U(\mathcal{A}_1, \ldots, \mathcal{A}_m) \). Suppose \( T(x) = i \) and \( f(x) = j \) where \( i \neq j \).

If \( x \in [\mathcal{A}_k] \) with \( k \neq i \) then \( T(x) = i \) is an error and \( x \in E_U \). If \( x \in [\mathcal{A}_i] \) then \( x \in E_{\mathcal{A}_i} \). Thus \( \mu(T(x) \neq f(x)) \leq \gamma + \alpha \) and so \( D_{\gamma + \alpha}(f) \leq G_\gamma(\mathcal{A}_1, \ldots, \mathcal{A}_m) \).

Since \( \epsilon - \gamma = \alpha < \beta_0, \gamma > \epsilon - \beta_0 = \epsilon_0 \), and thus \( \gamma + (\gamma/(3b))^3 > \epsilon \). This implies \( \alpha = \epsilon - \gamma < (\gamma/(3b))^3 \) and

\[
D_\epsilon(f) = D_{\gamma + \alpha}(f) \leq G_\gamma(\mathcal{A}_1, \ldots, \mathcal{A}_m).
\]

Combining this with (9) gives the result.
3 Conclusion and Notes

It would be interesting to resolve Conjecture 1.6. Unfortunately the analysis of Theorem 1.5 is not sufficient as the constant $K$ there goes to infinity as $\epsilon \to 0.$

It is natural to suppose

**Conjecture 3.1** Theorem 1.2 holds for all $f$ and for all $\alpha < \epsilon$ and Theorem 1.5 holds for all $A, B$ with $\delta(A, B) < \epsilon.$

In [12] Tardos noted that the following result from communication complexity is analogous to Theorem 1.1.

**Theorem 3.2** [1, 5] For all $X, Y$ finite and for all $f : X \times Y \to \{0, 1\},$

$$D^{cc}(f) \leq (N^{cc}_1(f) + 1)(N^{cc}_0(f) + 1).$$

Here, $D^{cc}(f)$ is the minimum depth of a two-party communication protocol for $f.$ For $j \in \{0, 1\},$ $N^{cc}_j(f)$ is the non-deterministic communication complexity of the decision problem, “Is $f(x, y) = j?$”. We have $N^{cc}_j(f) = \lceil \log_2(C^{cc}_j(f)) \rceil$ where $C^{cc}_j(f)$ is the minimum number of rectangles $X' \times Y' \subseteq f^{-1}(j)$ necessary to cover $f^{-1}(j).$ See [9] for more information.

It is natural to define approximate versions of these measures. Let $D^{cc}_\epsilon(f)$ be the minimum depth of a communication protocol for $f$ that is correct on
all but an $\epsilon$-fraction of inputs in $X \times Y$. This measure is used in distributional computational complexity. We define $N_{cc}^{j,\delta}(f) = \lceil \log_2(C_{cc}^{j,\delta}(f)) \rceil$, where $C_{cc}^{j,\delta}(f)$ is the minimum number of rectangles needed so that their union forms a set $S$ with $|S \triangle f^{-1}(j)| \leq \delta |X \times Y|$. It is natural to conjecture the following approximate version of Theorem 3.2:

**Conjecture 3.3** For all $\epsilon > 0$ there exists a $\beta_0 > 0$ such that for all $f : X \times Y \to \{0, 1\}$ with $X, Y$ finite and for all $\alpha_0, \alpha_1 \geq 0$, if $\alpha = \alpha_0 + \alpha_1 < \beta_0$ then

$$D_{cc}^{\epsilon}(f) \leq K N_{1,\alpha_1}^{cc}(f) N_{0,\alpha_0}^{cc}(f),$$

where $K$ depends only on $\epsilon$ and $\alpha$.

A simpler equivalent formulation is given by

$$\forall \epsilon > 0 \exists \alpha > 0 \exists K > 0 \forall f D_{cc}^{\epsilon}(f) \leq K N_{1,\alpha_1}^{cc}(f) N_{0,\alpha_0}^{cc}(f).$$

Let $g$ be $\epsilon$-close to $f$ if $f \neq g$ on at most a $\epsilon$-fraction of inputs. Let $N^{cc}(f) = \max\{N_{1}^{cc}(f), N_{0}^{cc}(f)\}$. If we define

$$N_{\epsilon}^{cc}(f) := \min\{N^{cc}(g) : g \text{ is } \epsilon\text{-close to } f\}$$

then trivially, $D_{\epsilon}^{cc}(f) = O((N_{\epsilon}^{cc}(f))^2)$. However, the above question, in which the rectangle covers are decoupled (i.e. are viewed as not necessarily coming from a single function $g$) does not appear to have been considered in the literature.
Acknowledgments

I would like to thank Mike Saks, Steven Rudich, and Russell Impagliazzo for many helpful discussions. In particular, the technique of “expansions” used in the proof of Theorem 1.5 was modeled after a technique developed by Rudich and Impagliazzo for related problems [7]. I would also like to thank the two referees for their insightful and helpful comments.

References


[12] Gabor Tardos, *Query complexity, or why is it difficult to separate* $\text{NP}^A \cap \text{co-NP}^A$ *from* $\text{P}^A$ *by random oracles* $A$?, *Combinatorica* 9 (1989), no. 4, 385–392. MR MR1054014 (91h:68070)