Conflict complexity is lower bounded by block sensitivity

Yaqiao Li*

October 23, 2018

Abstract

We show conflict complexity of any total boolean function, recently defined in [11] to study a composition theorem of randomized decision tree complexity, is at least a half of its block sensitivity. We also raise an interesting conjecture relating the composition theorem of randomized decision tree complexity to the long open conjecture that decision tree complexity is at most square of block sensitivity up to a constant.

Let $f : \{0,1\}^n \rightarrow \{0,1\}$ be a boolean function. The conflict complexity, denoted by $\chi(f)$, is a complexity measure of a boolean function that is recently defined in [11], and appears implicitly in [4]. Using this notion, [11] and [4] independently show a composition theorem for randomized decision tree complexity. Let $R(f)$ denote the randomized decision tree complexity of $f$ with bounded error. It is proven in [11] that $R(f \circ g) = \Omega(R(f) \sqrt{R(g)})$ for any relation $f$ and partial function $g$.

Another important complexity measure of a boolean function $f$ is its block sensitivity, denoted by $bs(f)$, is firstly defined in [9]. Block sensitivity, its variants, and related famous sensitivity conjecture are widely studied in complexity theory and combinatorics, see e.g. [10, 7, 6, 5, 1, 8] etc. The relation of block sensitivity to other complexity measures (e.g., decision tree complexity, approximate degree, etc), is relatively well understood, in particular most of them are polynomially related, see the survey [3].

In [11] it is shown that $\chi(f) = \Omega(R(f))$. Let $D(f)$ denote the deterministic decision tree complexity of $f$. Later after we formally define $\chi(f)$, it will be clear that $\chi(f) \leq D(f)$. Hence $\Omega(R(f)) \leq \chi(f) \leq D(f)$. Since $bs(f)$ is polynomially related to $D(f)$ and $R(f)$, as a result it is also polynomially related to $\chi(f)$. We show an explicit connection between them.

*McGill university. yaqiao.li@mail.mcgill.ca
Theorem 1. For any non-constant total boolean function \( f : \{0, 1\}^n \to \{0, 1\} \),

\[
(bs(f) + 1)/2 \leq \chi(f) \leq bs(f)^3.
\]

The upper bound follows trivially from known result. Below we formally define the conflict complexity and block sensitivity in Section 1, then we prove Theorem 1 in Section 2. We end with an interesting conjecture about composition theorem of randomized decision tree complexity and its relation to block sensitivity.

1 Conflict complexity and block sensitivity

Given a boolean function \( f : \{0, 1\}^n \to \{0, 1\} \), given \( \mu_0, \mu_1 \) be two distributions on \( f^{-1}(0) \) and \( f^{-1}(1) \), respectively. Given \( T \) as a deterministic decision tree that computes \( f \) correctly without error. For any node \( v \in T \), a basic fact is that \( v \) corresponds to a subcube of \( \{0, 1\}^n \) that is uniquely determined by the path leading from the root of \( T \) to \( v \). Denote

\[
\mu_0^v := \mu_0|_v, \quad \mu_1^v := \mu_1|_v.
\]

That is, they are the distributions of \( \mu_0 \) and \( \mu_1 \) conditioned on the subcube corresponding to \( v \).

Let \( x_v \) denote the variable at node \( v \), where the tree \( T \) branches to left or right according to \( x_v = 0 \) or \( x_v = 1 \), respectively. Denote

\[
\alpha_v := \Pr_{x \sim \mu_0^v} [x_v = 0], \quad \beta_v := \Pr_{x \sim \mu_1^v} [x_v = 0],
\]

where \( x \sim \mu_0^v \) means to sample \( x \in \{0, 1\}^n \) according to the distribution \( \mu_0^v \).

Consider the following random walk on the tree \( T \) as follows: at node \( v \),

- go to the left child (where \( x_v = 0 \)) with probability \( \min\{\alpha_v, \beta_v\} \);
- go to the right child (where \( x_v = 1 \)) with probability \( 1 - \max\{\alpha_v, \beta_v\} \);
- stop with probability \( |\alpha_v - \beta_v| \).

Figure 1 illustrates the case \( \alpha_v < \beta_v \): at node \( v \), the random walk branches to the left with probability \( \alpha_v \), to the right with probability \( 1 - \beta_v \), and stops with probability \( \beta_v - \alpha_v \).

\[\text{Footnote 1} \text{For more background on the definition of decision tree and related complexities, see the survey 3.}\]
Figure 1: Branching probability at node $v$

![Diagram of branching probability]

Note that the random walk depends on distributions $\mu_0$, $\mu_1$, and the tree $T$. By definition this random walk always goes forward (i.e., the direction from the root to leaves) along the tree $T$. We say a node $v \in T$ is a leaf if after the value of $x_v$ is queried, the tree must output accordingly. It is easy to see that $|\alpha_v - \beta_v| = 1$ if $v$ is a leaf, i.e., at a leaf the random walk always stops. We will be interested in the expected stopping time of the random walk.

**Definition 1 ([11]).** Given a boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, let $\mu_0, \mu_1$ denote distributions on $f^{-1}(0)$ and $f^{-1}(1)$, respectively. Let $T$ denote a deterministic decision tree that computes $f$ correctly without error. Define a random walk, as above, depending on $\mu_0, \mu_1$, and $T$.

Let $X := X(\mu_0, \mu_1, T)$ denote the random variable taking values in $\mathbb{N} = \{1, 2, 3, \ldots\}$ that represents the number of nodes the random walk has visited when it stops. The conflict complexity of $f$, denoted by $\chi(f)$, is defined as,

$$\chi(f) := \max_{\mu_0, \mu_1} \min_T \mathbb{E}_{\mu_0, \mu_1, T} X. \quad (3)$$

Since the random walk always stops if it reaches a leaf, we have $\chi(f) \leq D(f)$.

Next we define the block sensitivity. Let $x \in \{0, 1\}^n$, and $B \subseteq [n]$ be a subset. Denote $x^B$ as the $n$-bit string obtained from $x$ by flipping all bits in the subset $B$. For a given total function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, a subset $B$ is said to be a sensitive block for $x$ if $f(x) \neq f(x^B)$. Let $bs(f, x)$ denote the maximal number of disjoint sensitive blocks of $x$.

**Definition 2.** Given a total boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$, define the block sensitivity $bs(f)$ as $bs(f) = \max_x bs(f, x)$.

## 2 Proof of Theorem 1

To prove the lower bound of Theorem 1 it suffices to exhibit a distribution $\mu_0, \mu_1$, such that for any deterministic decision tree $T$ that computes $f$ correctly, one has $\mathbb{E}_{\mu_0, \mu_1, T} X \geq (bs(f) + 1)/2$.

**Proof of Theorem 1.** The upper bound follows as $\chi(f) \leq D(f) \leq bs(f)^3$, see the latter inequality in [3].
Next to show the lower bound. Denote $k = \text{bs}(f) \geq 1$ as we assume $f$ is not a constant function. Let $z \in \{0,1\}^n$ be an input string that achieves the block sensitivity of $f$, and $B_1, \ldots, B_k \subseteq [n]$ are the disjoint sensitive blocks of $z$. Denote $y_i = z^{B_i}$ for $i = 1, \ldots, k$. Without loss of generality assume $f(z) = 0$, then $f(y_1) = \cdots = f(y_k) = 1$.

Let $\mu_0$ be the distribution that is supported on the single point $z$, and $\mu_1$ be the uniform distribution over $Y = \{y_1, \ldots, y_k\}$. That is,

$$\mu_0(z) = 1, \quad \mu_1(y_i) = 1/k, \quad i = 1, \ldots, k.$$ 

Let $\mathcal{T}$ be any deterministic decision tree that computes $f$ correctly without error, and $X$ be the random variable defined as in Definition 1. Our aim is to show $\mathbb{E}_{\mu_0,\mu_1,\mathcal{T}} X \geq k/2$.

Since $\mathcal{T}$ is a deterministic decision tree, the input $z$ follows a specific path $P$ in $\mathcal{T}$. Let $\ell$ denote the length (i.e., number of nodes) of $P$, obviously $\ell \geq k$ since at least one bit from each sensitive block of $z$ must be queried in order to determine $f(z)$ correctly.

Recall the definition of $\mu_0^v$ and $\alpha_v$ given in (1) and (2), respectively. Since $\mu_0(z) = 1$, for any node $v \in P$, it is easy to see that $\mu_0^v(z) = 1$, i.e., the conditional distribution $\mu_0^v = \mu_0$ for any $v \in P$. Hence $\alpha_v = 1$ if the path $P$ branches to the left at node $v$, and $\alpha_v = 0$ if $P$ branches to the right.

By renaming variables if necessary, assume $x_1, x_2, \ldots, x_\ell$ are the successive nodes in the path $P$ where $x_1$ is the root and $x_\ell$ is the leaf. Here a node $v = x_i$ means to query the $i$-th variable of the input string.

It is illuminating to firstly analyze the branching probability at the root $v = x_1$. There are two cases:

- $1 \notin \bigcup_{j=1}^k B_j$. This implies $y_{i,1} = z_1$ for all $y_i \in Y$. Alternatively $\beta_v = \alpha_v$. Hence in this case,

$$\Pr[X = 1] = 0.$$ 

Also, $\Pr[\text{the random walk reaches } x_2] = 1$, and $\mu_1^{x_2} = \mu_1$.

- $1 \in B_j$ for some $j \in \{1,2,\ldots,k\}$. Since $B_j$ are disjoint from each other, there is at most one such $B_j$. In this case, without loss of generality assume $z_1 = 0$, hence $\alpha_v = 1$. Then $y_{j,1} = 1$ and $y_{i,1} = 0$ for all other $i \neq j$. Alternatively, $y_j$, and only $y_j$, will deviate from the path $P$. Hence,

$$\beta_v = \Pr_{x \sim \mu_1} [x_1 = 0] = \Pr_{y_i \sim Y} [y_{i,1} = 0] = (k - 1)/k.$$  

4
Therefore,
\[
\Pr[X = 1] = \alpha_v - \beta_v = 1/k.
\]

Also, \(\Pr[\text{the random walk reaches } x_2] = (k - 1)/k\), and \(\mu_1^{x_2}\) is a uniform distribution over \(Y - y_j\), a set of \(k - 1\) elements.

We claim that the above phenomenon is true in general: for any \(r = 1, \ldots, \ell\), either \(\Pr[X = r] = 0\) or \(\Pr[X = r] = 1/k\). Since \(\ell \geq k\), this immediately implies
\[
\mathbb{E} X \geq \sum_{i=1}^{k} i \cdot (1/k) = (k + 1)/2,
\]
as desired.

Now we show the claim. Consider a general node \(v = x_r \in P\). Let
\[
\mathcal{A}^v = \{ j : B_j \cap \{1, \ldots, r - 1\} = \emptyset \}, \quad Y^v = \cup_{j \in \mathcal{A}^v} \{y_j\}.
\]

Intuitively, \(Y^v\) is the set of \(y_j\) that are still “active” at node \(v\) (not deviated from the path \(P\) before \(v\)). By a similar analysis as we did for the root, we know that \(\mu_1^v\) is a uniform distribution on \(Y^v\). Hence, at node \(v = x_r\), the random walk stops with probability,
\[
|\alpha_v - \beta_v| = \begin{cases} 
0, & r \not\in \mathcal{A}^v; \\
1/|\mathcal{A}^v|, & r \in \cup_{j \in \mathcal{A}^v} B_j.
\end{cases}
\tag{4}
\]

On the other hand, for any two successive nodes \(x_i, x_{i+1}\) in \(P\),
\[
\Pr[\text{the random walk branches from } x_i \text{ to } x_{i+1}] = \begin{cases} 
1, & i \not\in \cup_{j \in \mathcal{A}^{x_i}} B_j; \\
(|\mathcal{A}^{x_i}| - 1)/|\mathcal{A}^{x_i}|, & i \in \cup_{j \in \mathcal{A}^{x_i}} B_j.
\end{cases}
\]

This implies
\[
\Pr[\text{the random walk reaches } v] = |\mathcal{A}^v|/k. \tag{5}
\]

Apply (4) and (5), we get \(\Pr[X = r] = \Pr[\text{the random walk reaches } v] \cdot |\alpha_v - \beta_v|\), is either 0 or 1/k as claimed.

3  A conjecture

In \cite{ref11}, it is shown for any relation \(f\) and total function \(g\), \(R(f \circ g) = \Omega(R(f) \cdot \chi(g))\), combine this with \(\chi(g) = \Omega(R(g))\) shows \(R(f \circ g) = \Omega(R(f) \cdot \sqrt{R(g)})\). By Theorem \cite{ref12} \(R(f \circ g) = \Omega(R(f) \cdot \text{bs}(g))\). This
result is already proven in [2]. In fact, [2] defines another complexity measure of a boolean function $g$ called sabotaged complexity, denote it by $RS(g)$. It is shown in [2], $R(f \circ g) = \Omega(R(f) \cdot RS(g))$ and $RS(g) \geq bs(g)/4$, hence it also implies $R(f \circ g) = \Omega(R(f) \cdot bs(g))$. Note that both $RS(g)$ and $\chi(g)$ are lower bounded by $bs(g)$.

**Conjecture 1.** For any relation $f$, and total function $g$,

$$R(f \circ g) = \Omega(R(f) \cdot \sqrt{D(g)}).$$

This immediately follows from the widely believed conjecture $D(g) = O(bs(g)^2)$, see [3].

In [2], it also shows $RS(g) = \Omega(\sqrt{R_0(g)/\log R_0(g)})$ where $R_0(g)$ denotes the randomized decision tree complexity of $g$ without error. Combine this with $R(f \circ g) = \Omega(R(f) \cdot RS(g))$, [2] shows $R(f \circ g) = \Omega(R(f) \cdot \sqrt{R_0(g)/\log R_0(g)})$. A positive answer to Conjecture 1 would improve both the two lower bounds (i.e., $R(f) \cdot \sqrt{R_0(g)/\log R_0(g)}$ and $R(f) \cdot \sqrt{R(g)}$) in the composition theorems from [2, 4, 11]. On the other hand, a negative answer to the Conjecture 1 would imply the long standing conjecture $D(g) = O(bs(g)^2)$ is false.

**Acknowledgement**

The author thanks the host of Simons Institute for the Theory of Computing where this work is done.

**References**

[1] Andris Ambainis, Mohammad Bavarian, Yihan Gao, Jieming Mao, Xiaoming Sun, and Song Zuo. Tighter relations between sensitivity and other complexity measures. In *International Colloquium on Automata, Languages, and Programming*, pages 101–113. Springer, 2014.

[2] Shalev Ben-David and Robin Kothari. Randomized query complexity of sabotaged and composed functions. *Theory OF Computing*, 14(5):1–27, 2018.

[3] Harry Buhrman and Ronald De Wolf. Complexity measures and decision tree complexity: a survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[4] Dmitry Gavinsky, Troy Lee, and Miklos Santha. On the randomised query complexity of composition. *arXiv preprint arXiv:1801.02226*, 2018.
[5] Mika Göös and Toniann Pitassi. Communication lower bounds via critical block sensitivity. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 847–856. ACM, 2014.

[6] Pooya Hatami, Raghav Kulkarni, and Denis Pankratov. Variations on the sensitivity conjecture. *arXiv preprint arXiv:1011.0354*, 2010.

[7] Claire Kenyon and Samuel Kutin. Sensitivity, block sensitivity, and -block sensitivity of boolean functions. *Information and Computation*, 189(1):43–53, 2004.

[8] Raghav Kulkarni and Avishay Tal. On fractional block sensitivity. *Chicago J. Theor. Comput. Sci.*, 2016:2, 2016.

[9] Noam Nisan. Crew prams and decision trees. *SIAM Journal on Computing*, 20(6):999–1007, 1991.

[10] David Rubinstein. Sensitivity vs. block sensitivity of boolean functions. *Combinatorica*, 15(2):297–299, 1995.

[11] Swagato Sanyal. A composition theorem via conflict complexity. *arXiv preprint arXiv:1801.03285*, 2018.