Decision Trees, Protocols, and the Fourier Entropy-Influence Conjecture

Andrew Wan*
Harvard University
atw12@seas.harvard.edu

John Wright†
Carnegie Mellon University
jswright@cs.cmu.edu

Chenggang Wu‡
IIIS,Tsinghua University
wcg06@mails.tsinghua.edu.cn

August 23, 2013

Abstract

Given $f : \{-1,1\}^n \to \{-1,1\}$, define the *spectral distribution* of f to be the distribution on subsets of $[n]$ in which the set S is sampled with probability $\widehat{f}(S)^2$. Then the *Fourier Entropy-Influence (FEI) conjecture* of Friedgut and Kalai [FK96] states that there is some absolute constant C such that $\mathbf{H}[\widehat{f}^2] \leq C \cdot \mathbf{Inf}[f]$. Here, $\mathbf{H}[\widehat{f}^2]$ denotes the Shannon entropy of f's spectral distribution, and $\mathbf{Inf}[f]$ is the total influence of f. This conjecture is one of the major open problems in the analysis of Boolean functions, and settling it would have several interesting consequences.

Previous results on the FEI conjecture have been largely through direct calculation. In this paper we study a natural interpretation of the conjecture, which states that there exists a communication protocol which, given subset S of $[n]$ distributed as $\widehat{f}^2$, can communicate the value of S using at most $C \cdot \mathbf{Inf}[f]$ bits in expectation. Using this interpretation, we are able show the following results:

- First, if f is computable by a read-k decision tree, then $\mathbf{H}[\widehat{f}^2] \leq 9k \cdot \mathbf{Inf}[f]$.
- Next, if f has $\mathbf{Inf}[f] \geq 1$ and is computable by a decision tree with expected depth d, then $\mathbf{H}[\widehat{f}^2] \leq 12d \cdot \mathbf{Inf}[f]$.
- Finally, we give a new proof of the main theorem of O'Donnell and Tan [OT13], i.e. that their FEI$^+$ conjecture composes.

In addition, we show that natural improvements to our decision tree results would be sufficient to prove the FEI conjecture in its entirety. Thus, our results are effectively the "last word" on the FEI conjecture as it relates to decision trees. We believe that our methods give more intuitive and illuminating proofs than previous results about the FEI conjecture.

*This research was partially supported by NSF grant CCF-964401.On leave from Tsinghua University.

†Supported by NSF grants CCF-0747250 and CCF-1116594 and a grant from the MSR–CMU Center for Computational Thinking.

‡This work was supported in part by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61061130540. Research done while visiting Carnegie Mellon University.

1 Introduction

Given a Boolean function $f : \{-1,1\}^n \to \{-1,1\}$, define the *spectral distribution* of f to be the distribution on subsets of $[n]$ in which the set S is sampled with probability $\widehat{f}(S)^2$. Overloading notation, we will denote this distribution by $\widehat{f}^2$. Write $\mathcal{X} \sim \widehat{f}^2$ for the random variable which is distributed according to $\widehat{f}^2$. The *Fourier Entropy-Influence (FEI) Conjecture* of Friedgut and Kalai [FK96] states that there is some absolute constant C such that $\mathbf{H}[\mathcal{X}] \leq C \cdot \mathbf{Inf}[f]$, where $\mathbf{Inf}[f]$ is the total influence of f, and $\mathbf{H}[\mathcal{X}]$ is the spectral entropy of f (equivalently, the Shannon entropy of $\mathcal{X}$), which equals

$$\mathbf{H}[\mathcal{X}] = \sum_{S \subseteq [n]} \widehat{f}(S)^2 \log\left(\frac{1}{\widehat{f}(S)^2}\right).$$

The FEI Conjecture has been shown to have several interesting consequences, including a learning algorithm for DNFs in the agnostic learning model [Man94, GKK08] , the famed KKL Theorem [KKL88], and resolving it is a central question in the analysis of Boolean functions. See [Kal07, OWZ11] for a comprehensive introduction to the subject.

Verifying the conjecture for individual functions—such as Majority, AND/OR, and Tribes(see [O'D03] for definitions)—can be done via straightforward calculation. Verifying it for larger classes of functions requires more subtle argumentation. To date, it has been shown to hold for random DNFs [KLW10], symmetric functions and read-once decision trees [OWZ11], and read-once formulas [OT13, CKLS13]. Unfortunately, this conjecture lends itself to proofs which are at times opaque and conceptually unilluminating. Perhaps one of the reasons is that whereas the total influence $\mathbf{Inf}[f]$ is a central quantity in the analysis of Boolean functions, the spectral entropy $\mathbf{H}[\mathcal{X}]$ is rarely encountered and poorly understood.

In this paper we consider the natural interpretation of the FEI conjecture as stating the existence of a coding scheme for the random variable $\mathcal{X}$ with a certain performance. Roughly speaking, the coding scheme must use, on average, some fixed constant times the *size* of $\mathcal{X}$ (see Section 1.1 for a precise description). Using this interpretation, we give three results concerning the FEI conjecture; we believe our proofs demonstrate that, for many cases, this interpretation could be the more natural way to reason about spectral entropy and can yield straightforward and conceptually interesting proofs.

For our first result, we verify the conjecture for read-k decision trees, where k is a constant. This is the class of decision trees in which each variable is queried at no more than k distinct locations in the entire tree. Previous results—those for read-once decision trees [OWZ11] and read-once formulas [OT13, CKLS13]—failed to generalize even to the read-twice case, as allowing a decision tree to be read-twice introduces correlations between different parts of the tree, and this is difficult to analyze. In this paper, we surmount this barrier, proving:

Theorem 1.1. *Suppose $f : \{-1,1\}^n \to \{-1,1\}$ can be computed by a read-k decision tree, and let $\mathcal{X} \sim \widehat{f}^2$. Then $\mathbf{H}[\mathcal{X}] \leq 9k \cdot \mathbf{Inf}[f]$.*

A natural question is whether this can be improved to show the FEI conjecture for read-$k(n)$ decision trees, where $k(n) = \omega(1)$ is a slowly growing function of n. However, a simple padding argument shows that this would be sufficient to prove the full FEI conjecture: given $f : \{-1,1\}^n \to \{-1,1\}$, one could add enough dummy variables to f so that $k(\cdot)$ is at least 2^n, and since any n-variable function is trivially computable by a read-2^n decision tree, f would satisfy the conjecture. Thus, it is natural that one should be able to prove the conjecture for read-constant decision trees but go no further.

Using much of the same proof as for Theorem 1.1, we then verify the conjecture for decision trees with expected depth d, where d is a constant. The FEI conjecture trivially holds for depth-d decision trees, which have a bounded number of variables, and so what makes this interesting is that we only require a bound on the *expected* depth of the tree. Our result is:

Theorem 1.2. *Suppose $f : \{-1,1\}^n \to \{-1,1\}$ is computable by a decision tree whose expected depth is d. Further, suppose $\mathbf{Inf}[f] \geq 1$. Then $\mathbf{H}[\widehat{f}^2] \leq 12d \cdot \mathbf{Inf}[f]$.*

As before, if we could show the FEI conjecture for decision trees with expected depth $d(n)$, where $d(n) = \omega(1)$ is a slowly growing function of n, we would be able to show the full FEI conjecture. In addition, the requirement in Theorem 1.2 that $\mathbf{Inf}[f]$ be reasonably large is necessary, as we show in Appendix E. We note that this result (with a better constant) also follows from the bound $\mathbf{H}[\widehat{f}^2] \leq 2d$, which was proven independently by [CKLS13].

For our final result, we give a new proof of the main theorem from [OT13]. Let f and $g_1, \ldots, g_k$ be Boolean functions, and consider the composition $h = f(g_1, \ldots, g_k)$, where each g_i is over its own set of variables. Their paper considers the following question: supposing that f and g_i's satisfy the FEI conjecture with constant C, what can one conclude about h? Perhaps their main contribution is in noting that from f's perspective, it is not receiving perfectly unbiased bits as inputs, but $\mathbf{E}[g_i]$-biased bits. Thus it is natural that it shouldn't matter whether f satisfies the FEI conjecture, but rather whether it satisfies some $\mathbf{E}[g_i]$-biased version of the FEI conjecture. They formulate this biased version of the FEI conjecture, which they call the FEI$^+$ conjecture [1], and prove the following composition theorem:

Theorem 1.3 (Informal)**.** *Suppose f and $g_1, \ldots, g_k$ satisfy the FEI$^+$ conjecture with constant C. Then $h = f(g_1, \ldots, g_k)$ also satisfies the FEI$^+$ conjecture with constant C.*

They proved this by expanding the expressions $\mathbf{H}[\mathcal{X}]$ and $\mathbf{Inf}[h]$ in terms of the Fourier coefficients of f and $g_1, \ldots, g_k$, and comparing the results. Owing to our interpretation of the FEI conjecture as being about codes, our new proof of this theorem shows that codes compose in a very clean way.

We now describe our interpretation of the FEI conjecture and discuss our main results in more detail.

1.1 The FEI Conjecture as a Coding Bound

Let $\mathcal{X} \sim \widehat{f}^2$. We view the Fourier Entropy-Influence Conjecture as stating the existence of highly efficient coding schemes for communicating the value of $\mathcal{X}$. To explain this, we begin with some standard information theory background. Given a domain $\mathcal{D}$ and an output alphabet Σ, a *code* on $\mathcal{D}$ is a function $c : \mathcal{D} \to \Sigma^*$. We say that c is *prefix-free* if $c(x)$ is never a prefix of $c(y)$ for distinct $x, y \in \mathcal{D}$. If $\boldsymbol{x}$ is a random variable which takes values in $\mathcal{D}$, then the average number of characters output by c, called the *length* of c, is $\mathbf{E}[|c(\boldsymbol{x})|]$, and we often care about finding a code c which minimizes this quantity. The source coding theorem of Shannon says that $\mathbf{H}[\boldsymbol{x}]$ is roughly the best possible length achievable by a prefix-free code:

Theorem 1.4 (Shannon's source coding theorem [Sha48])**.** *Let $\boldsymbol{x}$ be a random variable over a domain $\mathcal{D}$ and let Σ be a finite alphabet.*

1. *If $c : \mathcal{D} \to \Sigma^*$ is a prefix-free code for $\boldsymbol{x}$, then $\mathbf{H}[\boldsymbol{x}]/\log_2 |\Sigma| \leq \mathbf{E}[|c(\boldsymbol{x})|]$.*

[1] Another key aspect of their definition, necessary for composition to work, is that the functions f and $g_1, \ldots, g_k$ must satisfy a slightly stronger relationship between entropy and influence.

2. *Furthermore, there exists a prefix-free code* $c : \mathcal{D} \to \Sigma^*$ *such that* $\mathbf{E}[|c(\boldsymbol{x})|] \leq \mathbf{H}[\boldsymbol{x}]/\log_2 |\Sigma| + 1$.

(In fact, this theorem applies to the more general class of *uniquely decodable* codes, but it is sufficient for our purposes that we only consider prefix-free codes.)

This suggests that if we want to upper bound the entropy of $\mathcal{X} \sim \widehat{f^2}$, we should try to design an efficient protocol for communicating the value of $\mathcal{X}$. The formula $\mathbf{Inf}[f] = \sum_S |S| \cdot \widehat{f}(S)^2$ shows that $\mathbf{Inf}[f]$ is actually the expected size of the set $\mathcal{X}$. Thus, showing a bound of the form $\mathbf{H}[\mathcal{X}] \leq C \cdot \mathbf{Inf}[f]$ for a function f requires showing a protocol for communicating the value of $\mathcal{X}$ which uses at most a constant number of bits on average for each element of $\mathcal{X}$. As an example, consider the following protocol for encoding the value of a set $S \subseteq [n]$:

$\mathcal{P}(S)$**:**

- For each $i \in S$, output the $\lceil \log n \rceil$-bit description of i.
- Output $\perp$.

Here $\perp$ is a termination character which prevents different codewords from being prefixes of each other. (Without it, the codeword for $\{1\}$ would be a prefix of the codeword for $\{1, 2\}$, for example.)

Given the output of this protocol, one can uniquely determine the value of S. Furthermore, the protocol uses exactly $\lceil \log n \rceil \cdot |S| + 1$ characters to code S. As a result, we have

$$\mathbf{E}[|\mathcal{P}(\mathcal{X})|] = \lceil \log n \rceil \cdot \mathbf{E}[|\mathcal{X}|] + 1 = \lceil \log n \rceil \cdot \mathbf{Inf}[f] + 1, \tag{1}$$

giving an upper bound of $\mathbf{H}[\mathcal{X}] \leq \log_2 3 \cdot (\lceil \log n \rceil \cdot \mathbf{Inf}[f] + 1)$. This is (ignoring the $\log_2 3$ factor) the well-known "weak" upper bound [OWZ11, KMS12], which is essentially the best-known upper bound for a general Boolean f (and is tight when f is real-valued).

With some extra work, we can remove the $(+1)$ from Equation (1) while adding only a small factor to the coefficient of $\mathbf{Inf}[f]$. This is important for the case when f is heavily biased and $\mathbf{Inf}[f]$ is small (for example, when f is the AND function). As a start, consider the modified protocol $\mathcal{P}'$ which has the same first line as $\mathcal{P}$ but the following second line instead:

- If $S \neq \emptyset$, output $\perp$.

This will only output $\perp$ when $S \neq \emptyset$. For $\mathcal{X} \sim \widehat{f^2}$, the probability that $\mathcal{X} \neq \emptyset$ is $\sum_{S \neq \emptyset} \widehat{f^2}(S) = \mathbf{Var}[f] \leq \mathbf{Inf}[f]$. As a result, $\mathbf{E}[|\mathcal{P}'(\mathcal{X})|] \leq (\lceil \log n \rceil + 1) \cdot \mathbf{Inf}[f]$. However, $\mathcal{P}'$ is no longer prefix-free: $\mathcal{P}'(\emptyset)$ is the empty string, and is therefore a prefix of $\mathcal{P}'(S)$ for *every* S. The following lemma, which is implicit in [OWZ11], shows that such a protocol stills gives an entropy bound at a cost of $2 \cdot \mathbf{Inf}[f]$.

Lemma 1.5. *Let* $\mathcal{X} \sim \widehat{f^2}$*, and let* $\mathcal{P} : 2^{[n]} \to \Sigma^*$ *be a prefix-free protocol, except it outputs an empty string on the input* $\emptyset$*. Then* $\mathbf{H}[\mathcal{X}] \leq \log_2 |\Sigma| \cdot \mathbf{E}[|\mathcal{P}(\mathcal{X})|] + 2 \cdot \mathbf{Inf}[f]$.

For completeness, we include a proof of this lemma in Appendix B. Applying this lemma to the protocol $\mathcal{P}'$ in the previous example shows that $\mathbf{H}[\mathcal{X}] \leq (\log_2 3 \cdot (\lceil \log n \rceil + 1) + 2) \cdot \mathbf{Inf}[f]$.

As the above example illustrates, it is natural for a protocol to output nothing when $\mathcal{X} = \emptyset$. For convenience, we will call such protocols *almost* prefix-free. [2]

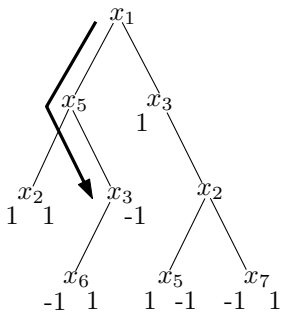


Figure 1: A path for the set $S = \{1, 3\}$. The other possible path is $x_1 \to x_3$.

1.2 Decision Tree Protocol

Let $f : \{-1, 1\}^n \to \{-1, 1\}$ be computed by a decision tree T, and let $\mathcal{X} \sim \widehat{f}^2$. To prove Theorems 1.1 and 1.2, we give an efficient protocol for communicating the value of $\mathcal{X}$. The protocol we use is simple: for a set $S \subseteq [n]$, $\widehat{f}(S)^2$ can be nonzero only if there is a root-to-leaf path in T which contains all the variables in S and, potentially, some extra variables. This means that any value which $\mathcal{X}$ takes with nonzero probability must correspond to at least one such path in the tree T. The protocol outputs the left/right description of such a path (stopping when the path has reached all the variables in $\mathcal{X}$), along with a sequence of bits indicating which indices along the path are contained in $\mathcal{X}$. Then, if $\mathcal{X} \neq \emptyset$, it terminates with a $\perp$.

For example, consider the tree in Figure 1. If the protocol were given the set $S = \{1, 3\}$, then there are two paths it could use: $x_1 \to x_5 \to x_3$ and $x_1 \to x_3$. Supposing it chose the first path, it would output $0, 1$ for the description of the path, then $1, 0, 1$ to indicate that x_1 and x_3 are in S but x_5 is not, and finally it would output $\perp$. So the total output string would be $0, 1, 1, 0, 1, \perp$. If it used the other path, the output string would be $1, 1, 1, \perp$. We defer the complete description of the protocol, including how it chooses between the possible paths, until Section 2.

Note that when $\mathcal{X} = \emptyset$, the protocol simply outputs an empty path. We show the following bound on the performance of this protocol which, when combined with Lemma 1.5 (and the fact that k and d are at least 1), yields Theorems 1.1 and 1.2:

Theorem 1.6. *Suppose $f : \{-1, 1\}^n \to \{-1, 1\}$ is computable by a read-k decision tree whose expected depth is d, and let $\mathcal{X} \sim \widehat{f}^2$. Then there is an almost prefix-free protocol for $\mathcal{X}$ with length at most $\min\{(2k + 2) \cdot \mathbf{Inf}[f], 4 \cdot \mathbf{Inf}[f] + 2d\}$ and alphabet size $|\Sigma| = 3$.*

This protocol relies heavily on the intuition that the structure of a decision tree should indicate which variables are significant. For example, the root variable should be very important, as should variables in the upper levels of the tree. Thus, even though the path outputted by the protocol always includes the root variable and and almost always includes the variables in the upper levels of the tree, this should not be a problem given that these variables are highly influential.

It is possible, however, to construct trees which do not fit this intuition: for example, consider a decision tree T which contains one set of variables on levels 0 through $l-1$, and has rooted at every node on level l a copy of a decision tree T' over a different set of variables. An example of such a tree is given in Figure 2 for $l = 2$. As all paths lead to T', the variables in the first l levels clearly have influence zero. Unfortunately, the described protocol will always output a path containing a variable from each of these l levels, as each path from the root to an influential variable must go

[2] An almost prefix-free protocol is implicit in the proof of the FEI conjecture for symmetric functions in [OWZ11], and ignoring the case when $\mathcal{X} = \emptyset$ is even explicitly built into the definition of the FEI$^+$ conjecture in [OT13].

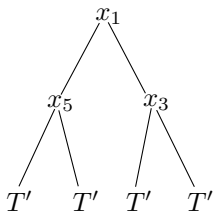


Figure 2: A bad tree. The T's are identical and do not contain x_1, x_3, or x_5.

through these levels. Thus, for any arbitrary l, one can make this protocol output $2l$ extraneous characters for any nonempty input set, regardless of the influence of the function.

This is not so problematic for the case when T has small expected depth or is read-k, for k a small constant. In the above example, every level of dummy variables adds one to the depth of T, and so this construction is limited by the expected depth of T. Furthermore, since a copy of T' is rooted at every level-l node, T is itself at least a read-2^l decision tree, in which case the fact that the protocol outputs only $2l$ more bits than it should is perhaps not too concerning.

To analyze this example, we note that for each level i between 0 and $l-1$, every node at level i has a pair of highly covariant children. In other words, for a node at level i, if g and h are the functions computed by that node's left and right subtrees, respectively, then as $g = h$, $\mathbf{Cov}[g, h] = \mathbf{Var}[f]$. Imagining that $\mathbf{Var}[f]$ is large, then it is exactly these nodes with highly covariant children which are troublesome. To keep track of these troublesome nodes, we define the quantity of *tree covariance* for T, written $\mathbf{Cov}[T]$. If T's left and right subtrees T_0 and T_1 compute the functions g and h, then $\mathbf{Cov}[T]$ can be defined recursively as $\mathbf{Cov}[T] = \mathbf{Cov}[g, h] + \frac{1}{2}(\mathbf{Cov}[T_0] + \mathbf{Cov}[T_1])$, with the base case that $\mathbf{Cov}[T] = 0$ if T computes a constant function. We show that the performance of this protocol on a general tree T depends on $\mathbf{Cov}[T]$:

Lemma 1.7. *The length of the above protocol is* $4 \cdot \mathbf{Inf}[f] + 2 \cdot \mathbf{Cov}[T]$.

It is a simple fact (see Proposition 2.2) that $\mathbf{Cov}[T] \leq d$ if T has expected depth d, and so Lemma 1.7 implies that the length of the protocol is at most $4 \cdot \mathbf{Inf}[f] + 2d$, which gives a part of Theorem 1.6.

Upper bounding $\mathbf{Cov}[T]$ for read-k decision trees is more complicated. For intuition, consider the case when $k = 2$. At the extreme, if $\mathbf{Cov}[g, h]$ were to equal one, then this would mean that $g = h$, in which case every variable relevant to g is also relevant to h, and vice versa. In particular, this means that every variable queried in T_0 to compute g must also be queried in T_1 to compute h, meaning that T_0 cannot have any variables which appear twice (as T is read-twice). And if T_0 is read-once, then the functions computed by its left and right subtrees must be entirely uncorrelated, as they depend on different variables. Thus, in this case $\mathbf{Cov}[T_0] = \mathbf{Cov}[T_1] = 0$, so $\mathbf{Cov}[T] = \mathbf{Cov}[g, h] = 1$. The result, intuitively, is that T has a finite amount of tree covariance to go around, and once it uses it up at a given level, the remaining levels must be uncorrelated. We extend this intuition into a bound on the tree covariance for read-k decision trees.

Lemma 1.8. *Let* $f : \{-1, 1\}^n \to \{-1, 1\}$ *be computed by a read-*k *decision tree* T*. Then* $\mathbf{Cov}[T] \leq (k-1) \cdot \mathbf{Var}[f]$.

Combining this lemma with Lemma 1.7 and the fact that $\mathbf{Var}[f] \leq \mathbf{Inf}[f]$ shows that the length of the protocol is at most $(2k+2) \cdot \mathbf{Inf}[f]$, giving the remaining part of Theorem 1.6.

1.3 Read-Once Composition Protocol

Theorem 1.3 from [OT13] shows that composing functions which satisfy the FEI$^+$ conjecture will result in a function which also satisfies FEI$^+$. We give a new proof of this theorem by proving an analogous result (Theorem 1.12 below) for *protocols* instead of entropy; our proof shows how to construct an efficient protocol for the composed function using the efficient protocols of each of the functions in the composition. To complete the proof of Theorem 1.3, which is a statement about entropies, one might try to use the source coding theorem to translate our result about protocols to a result about entropies. This can't be done so simply, however, as Theorem 1.4 only gives an approximate correspondence between protocols and entropy. We are able to get this step to work by using a (mostly standard) parallelizing technique. We now describe each of these two steps in more detail.

The FEI$^+$ conjecture works with the spectral distribution conditioned on a sample being non-empty. We write this distribution as $\mathcal{Y} \sim \widehat{f}^2 \setminus \emptyset$, which is defined so that:

$$\mathbf{Pr}[\mathcal{Y} = \emptyset] = 0, \quad \text{and} \quad \mathbf{Pr}[\mathcal{Y} = S] = \frac{\widehat{f}(S)^2}{1 - \widehat{f}(\emptyset)^2},$$

for any $S \neq \emptyset$. We assume here that $\widehat{f}(\emptyset)^2 < 1$ (the FEI Conjecture is trivial when $\widehat{f}(\emptyset)^2 = 1$). For our purposes, a prefix-free protocol $\mathcal{P}$ for $\mathcal{Y}$ is the same as an almost prefix-free protocol for $\mathcal{X} \sim \widehat{f}^2$: the equality $\mathbf{E}[|\mathcal{P}(\mathcal{X})|] = \mathbf{Var}[f] \cdot \mathbf{E}[|\mathcal{P}(\mathcal{Y})|]$ holds, and Lemma 1.5 tells us that we may obtain a bound on the entropy of $\mathcal{X}$ using a prefix-free protocol for $\mathcal{Y}$.

The FEI$^+$ conjecture in [OT13] stengthens the FEI conjecture and generalizes it to product distributions, making it amenable to composition. We use $\widetilde{f}$ to denote the Fourier transform of f with respect to a product distribution μ (here each bit x_i is set so that $\mathbf{E}_\mu[x_i] = \mu_i$). We now state the main definition from [OT13]:

Definition 1.9. Let $f : \{-1,1\}^n_\mu \to \{-1,1\}$ be a Boolean function. The function f satisfies FEI$^+$ with constant C if

$$\sum_{S \neq \emptyset} \widetilde{f}(S)^2 \log \left(\frac{\prod_{i \in S}(1 - \mu_i^2)}{\widetilde{f}(S)^2} \right) \leq C \cdot \sum_{S \neq \emptyset} \widetilde{f}(S)^2 (|S| - 1).$$

In [OT13], it was conjectured that for some constant C, every Boolean function satisfies FEI$^+$ with constant C. They were able to show that every Boolean function f satisfies FEI$^+$ with "constant" $2^{O(n)}$. [3]

Our first step is to reformulate what it means to "satisfy the FEI$^+$ conjecture with constant C" as a statement about the existence of an efficient protocol:

Definition 1.10. Let $f : \{-1,1\}^n_\mu \to \{-1,1\}$ be a function over the μ-biased variables $x_1, \ldots, x_n$, and let $\mathcal{Y} \sim \widetilde{f}^2 \setminus \emptyset$. Let P be a prefix-free protocol for communicating the value of $\mathcal{Y}$. Then P is a *C-good* protocol for f under bias μ if

$$\mathbf{E}[|P(\mathcal{Y})|] \leq C \cdot (\mathbf{E}[|\mathcal{Y}|] - 1) + \sum_i \mathbf{Pr}[i \in \mathcal{Y}] \cdot \log \frac{1}{1 - \mu_i^2} + \log \underset{p}{\mathbf{Var}}[f].$$

This definition can be derived by rearranging the inequality in Definition 1.9 to place $\sum_{S \neq \emptyset} \tilde{f}(S)^2 \log \frac{1}{\tilde{f}(S)^2}$ on the left-hand side, and then replacing $\sum_{S \neq \emptyset} \tilde{f}(S)^2 \log \frac{1}{\tilde{f}(S)^2} = \mathbf{H}[\mathcal{Y}]$ with $\mathbf{E}[|P(\mathcal{Y})|]$. Furthermore, because $\mathbf{H}[\mathcal{Y}] \leq \mathbf{E}[|P(\mathcal{Y})|]$, any function with a good protocol automatically satisfies FEI$^+$:

[3] It is known that one can improve this to $O(\log(n))$ in the unbiased case when all the μ_i's are zero.

Fact 1.11. *Suppose there exists a C-good protocol for f under bias μ. Then f satisfies (the μ-biased) FEI$^+$ with constant C.*

We then prove the following composition theorem for protocols in Section 3:

Theorem 1.12. *Let $h(x^1, \dots, x^k) = f(g_1(x^1), \dots, g_k(x^k))$, where the domain of h is endowed with a product distribution μ. Suppose there are C-good protocols for $g_1, \dots, g_k$ under μ and a C-good protocol for f under bias $\eta = \langle \mathbf{E}_\mu[g_1], \dots, \mathbf{E}_\mu[g_k] \rangle$. Then there exists a C-good protocol for h under bias μ.*

Given a good protocol P_f for f and good protocols $P_1, \dots, P_k$ for $g_1, \dots, g_k$, we construct a good protocol for h in the following way. Let $\mathcal{Y} = \mathcal{Y}_1 \circ \cdots \circ \mathcal{Y}_k$ be drawn from $\widetilde{h}^2 \setminus \emptyset$, where each $\mathcal{Y}_i$ denotes the restriction of $\mathcal{Y}$ to the relevant coordinates of g_i. Note that the $\mathcal{Y}_i$'s form a partition of $\mathcal{Y}$ because the g_i's have disjoint inputs. The protocol will use P_f to specify which $\mathcal{Y}_i$ are non-empty, and, for each such i, it will use $P_i(\mathcal{Y}_i)$ to specify which of the bits relevant to g_i are present in $\mathcal{Y}$. While outputting all of $P_1(\mathcal{Y}_1), \cdots, P_k(\mathcal{Y}_k)$ would be simpler and would suffice to completely specify $\mathcal{Y}$, this protocol will not be efficient when the g_i's have small variance (in this case the number of non-empty $\mathcal{Y}_i$ may be quite small).

In fact, the set $S \subseteq [k]$ of non-empty $\mathcal{Y}$ will be distributed according to $\widetilde{f}^2 \setminus \emptyset$, where $\widetilde{f}$ denotes the η-biased Fourier transformation of f, and furthermore, the sets $\mathcal{Y}_i$ are distributed according to $\widetilde{g_i}^2 \setminus \emptyset$. This fact is somewhat implicit in the analysis of [OT13], though we find it it somewhat clearer and simpler to prove in isolation, without reference to entropy. The analysis of this protocol follows almost immediately from this fact, as the protocols P_f and $P_1, \dots, P_k$ are designed for these distributions.

This yields a composition theorem for protocols. Our ultimate goal, however, is to prove the following composition theorem for FEI$^+$:

Theorem 1.13. *Let $h(x^1, \dots, x^k) = f(g_1(x^1), \dots, g_k(x^k))$, where the domain of h is endowed with a product distribution μ. Suppose $g_1, \dots, g_k$ satisfy μ-biased FEI$^+$ with constant C and f satisfies η-biased FEI$^+$ with constant C, where $\eta = \langle \mathbf{E}_\mu[g_1], \dots, \mathbf{E}_\mu[g_k] \rangle$. Then h satisfies μ-biased FEI$^+$ with constant C.*

The naive strategy would be to apply Shannon's source coding theorem to derive C-good protocols for $f, g_1, \dots, g_k$, apply Theorem 1.12 to give a C-good protocol for h, and then apply Fact 1.11 to show that h satisfies FEI$^+$. Unfortunately, this fails in the first step: the source coding theorem loses an additive factor of $(+1)$ when translating from entropy to protocols, and this $(+1)$ means that $f, g_1, \dots, g_k$ don't necessarily have C-good protocols.

To fix this problem, we use the well-known observation that the length of a protocol can be made arbitrarily close to the entropy of a given random variable by encoding many independent copies of that random variable. Thus, by switching to protocols which encode multiple copies of $\mathcal{Y}$ instead of just one, we can ensure that the first step goes through properly, and the other steps (such as Theorem 1.12 go through nearly identically in this setting as well. As this part of the argument is essentially standard, we sketch it briefly in Appendix C.

1.4 Organization

The decision tree results can be found in Section 2 , and the FEI$^+$ results can be found in Section 3. The appendices mostly contain proofs of simple lemmas. Appendix E contains the argument for why the restriction on the total influence of f in Theorem 1.2 is necessary.

Proofs of the main theorems. Theorem 1.1 and Theorem 1.2 follows from Lemma 1.5 and Theorem 1.6.Theorem 1.3 follows from Theorem 1.12(proved in Section 3) and Theorem 1.13.

2 Entropy-Influence for read-k decision trees

In this section, we analyze our communication protocol for decision trees . We begin with some preliminary definitions in Section 2.1. Then, as a simple first step, we consider the case of read-once decision trees in Section 2.2. Finally, we prove Lemma 1.7 in Section 2.3 and Lemma 1.8 in Section 2.4. Together, these prove Theorem 1.6.

2.1 Definitions and Notation

Fourier analysis. Any function $f : \{-1,1\}^n \to \mathbb{R}$ can be written as

$$f(x) = \sum_{S \subseteq [n]} \widehat{f}(S)\chi_S(x).$$

The $\widehat{f}(S)$'s are the *Fourier coefficients* of f, and for each $S \subseteq [n]$, the parity function χ_S is defined as $\chi_S(x) = \prod_{i \in S} x_i$. *Parseval's equation* will be important for us, which states that $\mathbf{E}_{\boldsymbol{x}}[f(\boldsymbol{x})^2] = \sum_S \widehat{f}(S)^2$. In particular, if f is ± 1-valued, then this sum equals one, and so the squared coefficients $\widehat{f}(S)^2$ form a probability distribution. We will also need the formula $\mathbf{Var}[f] = \sum_{S \neq \emptyset} \widehat{f}(S)^2$. We note that if $\mathcal{X} \sim \widehat{f}^2$, then $\mathbf{Pr}[\mathcal{X} \neq \emptyset] = \mathbf{Var}[f]$.

The *influence* of a variable x_i on f is $\mathbf{Inf}_i[f] := \mathbf{Pr}_{\boldsymbol{x}}[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{\oplus i})]$, where $\boldsymbol{x}^{\oplus i}$ is $\boldsymbol{x}$ with the i-th bit flipped. The *total influence* of f is $\mathbf{Inf}[f] := \sum_i \mathbf{Inf}_i[f]$, and it is simple to show that $\mathbf{Inf}[f]$ can also be written as $\mathbf{Inf}[f] = \sum_{S \neq \emptyset} |S| \widehat{f}(S)^2$. Comparing this to the formula for $\mathbf{Var}[f]$ shows that $\mathbf{Var}[f] \leq \mathbf{Inf}[f]$. This is all the Fourier analysis we will need; for a more comprehensive introduction to the subject, see [O'D].

Decision trees. Decision trees are a standard model of computation, and we omit their definition (see, for example, [OWZ11] for a definition). Given a tree T, we will call the subtree corresponding to the $+1$ edge the *left* subtree and the subtree corresponding to the -1 edge the *right* subtree. We will assume that if T is a decision tree, then no variable appears more than once in any root-to-leaf path of T. If this is not the case, then T can be simplified. We say that T is a *read-k* decision tree if no variable is queried in more than k locations of T.

Given a decision tree T, if v is a node of T, then $l(v)$ is the *label* of v, i.e. the coordinate in x which is queried at node v. Let $r(T)$ be the root node of T. Next, set $d(v)$ to be the depth of v in T. We start counting the depth at 0, so that $d(r(T)) = 0$. The *expected depth* of T is the average number of bits T queries on a uniformly random input $\boldsymbol{x}$. Since a given node v is reached with probability $2^{-d(v)}$, the expected depth of T may be written as

$$\sum_{v \in T} 2^{-d(v)}. \tag{2}$$

Now we may state our main definition:

Definition 2.1. Given a decision tree T and an internal node v, let g be the function computed by v's left subtree and h be the function computed by v's right subtree. Then define

- $\mathbf{Cov}[v] := \mathbf{Cov}[g, h]$,

- $\mathbf{Cov}_i[T] := \sum_{v:l(v)=i} \mathbf{Cov}[v] \cdot 2^{-d(v)}$, and
- $\mathbf{Cov}[T] := \sum_{v \in T} \mathbf{Cov}[v] \cdot 2^{-d(v)}$.

Note that $\mathbf{Cov}[T]$ may also be written as $\mathbf{Cov}[T] = \sum_{i \in [n]} \mathbf{Cov}_i[T]$. Furthermore, if T_0 is T's left subtree and T_1 is T's right subtree, then $\mathbf{Cov}[T]$ may also be written recursively as $\mathbf{Cov}[T] = \mathbf{Cov}[g, h] + \frac{1}{2}(\mathbf{Cov}[T_0] + \mathbf{Cov}[T_1])$, with the base case that $\mathbf{Cov}[T] = 0$ if T performs no queries. Intuitively, $\mathbf{Cov}[T]$ is a measure of the total correlation present in the structure of T. For example, $\mathbf{Cov}[T] = 0$ if T is a read-once decision tree. We note that when T computes a Boolean function, $\mathbf{Cov}[v] \leq 1$ for each $v \in T$. Thus, in this case, it is immediate from Equation (2) that the expected depth of T is at least $\mathbf{Cov}[T]$. This gives the following proposition.

Proposition 2.2. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by T, a decision tree with expected depth d. Then $\mathbf{Cov}[T] \leq d$.*

We will also need the following two propositions, which are proven in Appendix A.

Proposition 2.3. *Let f be computed by a decision tree T whose left and right subfunctions are g and h, respectively. If x_i is at the root of T and S is any subset of $[n] \setminus \{i\}$, then*

$$\widehat{f}(S)^2 + \widehat{f}(S \cup \{i\})^2 = \frac{1}{2}\left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right).$$

Proposition 2.4. *Assume the setup from Proposition 2.3. Then for a coordinate $j \neq i$,*

$$\mathbf{Inf}_j[f] = \frac{1}{2} \cdot (\mathbf{Inf}_j[g] + \mathbf{Inf}_j[h]).$$

2.2 Read-once decision trees

In this section, we will sketch the argument for read-once decision trees. Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by a read-once decision tree T. Given a decision tree T and a path $P = v_1 \to \cdots \to v_k$ in the tree (starting at the root v_1), the *description* of the path is the sequence of bits $b_1, \ldots, b_{k-1} \in \{0,1\}$ which, if read in that order, would result in traversing the given path (here we are using the standard $1 \leftrightarrow 0$ and $-1 \leftrightarrow 1$ correspondance). Given a set S, our protocol will output the description of a path in which S is a subset of $\{l(v_1), \ldots, l(v_k)\}$. In fact, our protocol will choose a *minimal* such path containing S, in the sense that the path will stop once it has encountered all of the variables in S. In a general decision tree, there could be many minimal paths containing S and starting at the root, but because T is read-once, there can only be one such path. We may therefore state the protocol as:

Given $S \subseteq [n]$:

1. If $S = \emptyset$, output nothing.
2. There is a minimal path $P = p_1 \to \cdots \to p_k$ containing the indices in S which starts at T's root.
3. Output the description of that path.
4. Output a bit sequence $b_1, \ldots, b_k \in \{0,1\}$, where $b_i = 1$ iff $p_i \in \mathcal{X}$.
5. Terminate with a $\perp$.

Suppose that the path P the protocol finds is of length l. Then because the description of a path of length l uses $l-1$ bits, the protocol outputs $2l$ characters in total. Furthermore, the protocol accurately communicates the value of S: given the output of the protocol, one could reconstruct S by following the path indicated by the first $l-1$ bits and including only those indices along the path which are tagged with a 1 in the second sequence. So long as $S \neq \emptyset$, the output is terminated with a $\perp$ character. Together, these mean that the protocol is an almost prefix-free protocol with alphabet size $|\Sigma| = 3$.

We are interested in the length of the protocol on input $\mathcal{X} \sim \widehat{f^2}$. As shown above, the number of characters this protocol outputs is exactly twice the length of the path P. Thus, we need to upper bound the average length of P.

Let us consider reasons why P might be on average too long. For example, because the protocol only considers paths starting at the root, the path output always contains the root variable (unless $\mathcal{X} = \emptyset$), even though this variable might have very low influence on f. However, a simple argument shows that this worry is unfounded. In particular, if x_i is T's root variable, then $\mathbf{Inf}_i[f] \geq \frac{1}{2}\mathbf{Var}[f]$ (we will show this later in Lemma 2.6). This inequality uses crucially the fact that T is read-once. The path P contains x_i whenever $\mathcal{X} \neq \emptyset$, which happens with probability $\mathbf{Var}[f]$. Thus, the probability that P contains x_i is at most $2 \cdot \mathbf{Inf}_i[f]$.

An inductive argument allows us to bring this inequality down to the rest of the variables in the tree, showing that the probability P contains a variable x_j is at most $2 \cdot \mathbf{Inf}_j[f]$ (we will show this later in Lemma 2.5). Summing this inequality over all j shows that the expected length of P is at most $2 \cdot \mathbf{Inf}[f]$. Thus, the protocol outputs at most $4 \cdot \mathbf{Inf}[f]$ characters in expectation, proving Theorem 1.7 in the $k = 1$ case.

2.3 General decision trees

Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by a decision tree T. Generalizing the above argument to work for T requires some modifications. The main change is that given $S \subseteq [n]$, there is no longer necessarily a unique minimal path starting from T's root which contains the indices in S. As Figure 1 shows, there could be two paths to select from when, for example, $S = \{1,3\}$. We want our protocol to use the fewest characters possible, so the obvious choice is for it to simply use the shortest path possible. This protocol is difficult to analyze, however, so we instead use a suboptimal protocol which constructs a path vertex-by-vertex probabilistically. If g is the function computed by T's left subtree and h is the function computed by T's right subtree, then the first step of the path will be chosen based on the relative weight that g and h place on the set S, i.e. $\widehat{g}(S)^2$ versus $\widehat{h}(S)^2$. As a result, the protocol is most easily stated recursively, as follows:

$\mathcal{P}(T, S)$**:**

1. If $S = \emptyset$, output nothing and terminate.
2. Let g be the function computed by T's left subtree T_0, and likewise let h be the function computed by T's right subtree T_1.
3. Let x_i be T's root variable. If $i \in S$, output a 1. Otherwise, output a 0.
4. Set $S' = S \setminus \{i\}$. If $S' = \emptyset$, output $\perp$ and terminate.
5. With probability proportional to $\widehat{g}(S')^2$, output 0 and run $\mathcal{P}(T_0, S')$.
6. With probability proportional to $\widehat{h}(S')^2$, output 1 and run $\mathcal{P}(T_1, S')$.

This protocol outputs the same information that the protocol in Section 2.2 does, only now the description of the path and the bit sequence are interleaved. If this protocol outputs $2k$ characters, then characters $2, 4, \ldots, 2k-2$ give a description of a path P, characters $1, 3, \ldots, 2k-1$ indicate which indices along the path P are included in S, and the $2k$-th character is a $\perp$. As a result, this protocol is an almost prefix-free protocol with alphabet size $|\Sigma| = 3$. We will refer to the path P as the path the protocol outputs, selects, etc.

Let us now consider the length of the protocol on input $\mathcal{X} \sim \widehat{f^2}$. The number of characters output is exactly twice the length of the path P the protocol outputs. Thus, we would like to upper bound the expected length of P. Our main lemma will show that for a given variable x_i, we can upper-bound the probability that it appears in P as follows:

Lemma 2.5. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by a decision tree T, let $\mathcal{X} \sim \widehat{f^2}$, and let $p_i(T)$ be the probability that the path selected by $\mathcal{P}(T, \mathcal{X})$ contains index i. Then $p_i(T) \leq 2 \cdot \mathbf{Inf}_i[f] + \mathbf{Cov}_i[T]$.*

By summing this lemma over $i \in [n]$, the expected length of P is at most $2 \cdot \mathbf{Inf}[f] + \mathbf{Cov}[T]$, and so the expected number of characters output by the protocol is at most $4 \cdot \mathbf{Inf}[f] + 2 \cdot \mathbf{Cov}[T]$, which proves Lemma 1.7.

In the special case when T has expected depth d, Proposition 2.2 tells us that $\mathbf{Cov}[T] \leq d$, so the protocol uses at most $4 \cdot \mathbf{Inf}[f] + 2 \cdot d$ characters in expectation, the bound given in Theorem 1.6. If we further assume that $\mathbf{Inf}[f] \geq 1$, then this quantity is at most $6d \cdot \mathbf{Inf}[f]$. Combining this with Lemma 1.5 yields our FEI bound for decision trees of expected depth d, Theorem 1.2. In Appendix E, we argue that proving this theorem without the restriction that $\mathbf{Inf}[f] \geq 1$ is unlikely so long as the Fourier Entropy-Influence conjecture remains unproven. Next, as upper-bounding $\mathbf{Cov}[T]$ is more involved if T is read-k, we will defer the proof of the FEI conjecture for read-k decision trees to Section 2.4.

Now we prove Lemma 2.5. In Section 2.2, we stated that if x_i is T's root variable, then $\mathbf{Inf}_i[f] \geq \frac{1}{2}\mathbf{Var}[f]$, supposing that T is read-once. Unfortunately, this is not true for general (or even read-twice) decision trees. For example, the root variable could have two identical subtrees as its children, in which case it has influence zero. For this to happen, though, it must be the case that the two subfunctions have high covariance.

Lemma 2.6. *Let $f : \{-1,1\}^n \to \{-1,1\}$ be computed by a decision tree T. If x_i is at the root of T, then $\mathbf{Inf}_i[f] \geq \frac{1}{2}\mathbf{Var}[f] - \frac{1}{2}\mathbf{Cov}[r(T)]$.*

Proof. Let g be the function computed by T's left subtree and h be the function computed by T's right subtree, so that $f(x) = g(x)$ if $x_i = 1$, and $f(x) = h(x)$ if $x_i = -1$. Then $\hat{f}(\emptyset)^2 = \left(\frac{\hat{g}(\emptyset)+\hat{h}(\emptyset)}{2}\right)^2$. As a result,

$$\begin{aligned}
\mathbf{Inf}_i[f] &= \mathbf{Pr}[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{\oplus i})] \\
&= \mathbf{Pr}[g(\boldsymbol{x}) \neq h(\boldsymbol{x})] \\
&= \frac{1}{2} - \frac{1}{2}\,\mathbf{E}[g(\boldsymbol{x})h(\boldsymbol{x})] \\
&= \frac{1}{2} - \frac{1}{2}\hat{g}(\emptyset)\hat{h}(\emptyset) - \frac{1}{2}\mathbf{Cov}[g,h] \\
&\geq \frac{1}{2} - \frac{1}{2}\left(\frac{\hat{g}(\emptyset)+\hat{h}(\emptyset)}{2}\right)^2 - \frac{1}{2}\mathbf{Cov}[g,h] && \text{(using } ab \leq \left(\tfrac{a+b}{2}\right)^2\text{)} \\
&= \frac{1}{2}\left(1 - \hat{f}(\emptyset)^2\right) - \frac{1}{2}\mathbf{Cov}[g,h] \\
&= \frac{1}{2}\,\mathbf{Var}[f] - \frac{1}{2}\mathbf{Cov}[g,h].
\end{aligned}$$

Because $\mathbf{Cov}[r(T)] = \mathbf{Cov}[g,h]$, this proves the lemma. □

We now use this to prove Lemma 2.5:

Proof of Lemma 2.5. We prove this by structural induction on the tree T, based on whether x_i is at the root of T. The lemma is clearly true if i doesn't appear in T, so we will assume that it does.

Base case: In this case, the root of T is x_i. By the protocol above, x_i will *always* be on the path P unless $\mathcal{X} = \emptyset$, i.e. the path is empty. Thus, the probability that x_i is outputted is $1-\hat{f}(\emptyset)^2 = \mathbf{Var}[f]$. By Lemma 2.6, we have that $2 \cdot \mathbf{Inf}[f] \geq \mathbf{Var}[f] - \mathbf{Cov}[r(T)] = p_i(T) - \mathbf{Cov}[r(T)]$. Because x_i is at the root, it can appear nowhere else in T. This means that $\mathbf{Cov}_i[T] = \mathbf{Cov}[r(T)]$, which concludes the base case.

Inductive step: In this case, the root of T is not x_i, meaning that x_i is queried in one (or both) of T's subtrees. Let T_0 be the left subtree of T and T_1 its right subtree, and assume without loss of generality that the root of T is x_n. We will show the following pair of simple equalities:

1. $\mathbf{Inf}_i[f] = \frac{1}{2} \cdot (\mathbf{Inf}_i[g] + \mathbf{Inf}_i[h])$, and
2. $p_i(T) = \frac{1}{2} \cdot (p_i(T_0) + p_i(T_1))$.

Equality 1 follows directly from Proposition 2.4. Before proving Equality 2, let's see how they imply the lemma.

$$\begin{aligned}
2 \cdot \mathbf{Inf}_i[f] &= \mathbf{Inf}_i[g] + \mathbf{Inf}_i[h] \\
&\geq \frac{1}{2}\left(p_i(T_0) + p_i(T_1) - \mathbf{Cov}_i[T_0] - \mathbf{Cov}_i[T_1]\right) \\
&= p_i(T) - \frac{1}{2}\left(\mathbf{Cov}_i[T_0] + \mathbf{Cov}_i[T_1]\right), \qquad (3)
\end{aligned}$$

where the second line follows from applying the inductive hypothesis to g and h. Since each vertex v in T_0 (or T_1) is one edge farther from the root in T than it is in T_0 (or T_1), we get that

$\mathbf{Cov}_i[T] = \frac{1}{2}\mathbf{Cov}_i[T_0] + \frac{1}{2}\mathbf{Cov}_i[T_1]$. Note that $\mathbf{Cov}[r(T)]$ doesn't contribute anything to $\mathbf{Cov}_i[T]$ because x_i is not at the root of T. Plugging this equality into Equation (3) yields the lemma.

Now, we prove Equality 2. It will be convenient for us to define the modified protocol $\mathcal{P}'$:

> $\mathcal{P}'(T, S)$**:**
>
> 1. If $S \neq \emptyset$ and $S \neq \{j\}$, where x_j is T's root variable, then run $\mathcal{P}(T, S)$.
> 2. Otherwise:
>   (a) If $S = \{j\}$, output the characters $1, \perp$.
>   (b) With probability proportional to $\widehat{g}(\emptyset)^2$, run $\mathcal{P}(T_0, \emptyset)$.
>   (c) With probability proportional to $\widehat{h}(\emptyset)^2$, run $\mathcal{P}(T_1, \emptyset)$.

Note that $\mathcal{P}'$ always calls $\mathcal{P}$ as a subroutine. When $S \neq \emptyset, \{j\}$, then $\mathcal{P}'(T, S)$ is identical to $\mathcal{P}(T, S)$. On the other hand, when S equals $\emptyset$ or $\{j\}$, then $\mathcal{P}'(T, S)$ outputs exactly what $\mathcal{P}(T, S)$ would output, but then it calls either $\mathcal{P}(T_0, \emptyset)$ or $\mathcal{P}(T_1, \emptyset)$. These two will immediately terminate, so $\mathcal{P}'$ has the same output behavior as $\mathcal{P}$. Thus, to show that $p_i(T) \leq 2 \cdot \mathbf{Inf}_i[f] + \mathbf{Cov}_i[T]$, it suffices to show that the probability that the path selected by $\mathcal{P}'(T, \cdot)$ contains index i is at most $2 \cdot \mathbf{Inf}_i[f] + \mathbf{Cov}_i[T]$.

We will show that the probability $\mathcal{P}'(T, \cdot)$ makes a call to $\mathcal{P}(T_0, \cdot)$ versus $\mathcal{P}(T_1, \cdot)$ is exactly $\frac{1}{2}$. Next, we will show that the sets it calls $\mathcal{P}(T_0, \cdot)$ with are distributed as $\widehat{g}^2$, and similarly for $\mathcal{P}(T_1, \cdot)$, so that the recursion works.

Without loss of generality, assume that x_n is the root variable of T. Let $S \subseteq [n-1]$ be any set. The protocol $\mathcal{P}'(T, \mathcal{X})$ can only call $\mathcal{P}(T_0, S)$ when $\mathcal{X}$ is either S or $S \cup \{n\}$, which happens with probability $\widehat{f}(S)^2 + \widehat{f}(S \cup \{n\})^2$. By Proposition 2.3, $\widehat{f}(S)^2 + \widehat{f}(S \cup \{n\})^2 = \frac{1}{2}\left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right)$. In either of these two cases, $\mathcal{P}(T_0, S)$ is called with probability proportional to $\widehat{g}(S)^2$, and $\mathcal{P}(T_1, S)$ is called with probability proportional to $\widehat{h}(S)^2$. Thus, the probability that $\mathcal{P}(T_0, S)$ is called is

$$\frac{1}{2}\left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right) \cdot \frac{\widehat{g}(S)^2}{\widehat{g}(S)^2 + \widehat{h}(S)^2} = \frac{\widehat{g}(S)^2}{2}.$$

Summing over all sets S, the probability that $\mathcal{P}(T_0, \cdot)$ is called is exactly $1/2$, and conditioned on this occurring, the probability that $\mathcal{P}(T_0, S)$ is called is exactly $\widehat{g}(S)^2$. A similar argument holds with T_1 in place of T_0 and h in place of g.

Thus, when $\mathcal{P}'(T, \mathcal{X})$ calls $\mathcal{P}(T_0, \cdot)$, the input to the recursive call is distributed as $\widehat{g}^2$, meaning that the path constructed in the recursive call contains x_i with probability $p_i(T_0)$. Similarly, when $\mathcal{P}'(T, \mathcal{X})$ calls $\mathcal{P}(T_1, \cdot)$, the path constructed in the recursive call contains x_i with probability $p_i(T_1)$. Combining these, $p_i(T) = \frac{1}{2}\left(p_i(T_0) + p_i(T_1)\right)$. □

2.4 A covariance bound for read-k decision trees

In this section, we prove Lemma 1.8.

Lemma 2.7 (Lemma 1.8 restated.)**.** *Let $f : \{-1, 1\}^n \to \{-1, 1\}$ be computed by a read-k decision tree T. Then $\mathbf{Cov}[T] \leq (k-1) \cdot \mathbf{Var}[f]$.*

Combining this with Lemma 1.7 and Lemma 1.5 yields our FEI bound for read-k decision trees:

Theorem 2.8. *If $f : \{-1,1\}^n \to \{-1,1\}$ is computable by a read-k decision tree, then* $\mathbf{H}[\widehat{f}^2] \leq (2 + (2k+2) \cdot \log_2 3) \cdot \mathbf{Inf}[f]$.

It is not at all clear whether our upper bound in Lemma 1.8 is tight. Potentially, this bound could be replaced with $\mathbf{Cov}[T] \leq \log_2 k \cdot \mathbf{Var}[f]$. The tight example of this was presented earlier: let T be the tree given in Figure 2, only with l layers of dummy variables rather than just two. Furthermore, suppose that T' is itself read-once. It is easy to see that T is read-2^l and has tree-covariance $\mathbf{Cov}[T] = l \cdot \mathbf{Var}[f]$. Thus, in this case, $\mathbf{Cov}[T] = \log 2^l \cdot \mathbf{Var}[f]$.

We will prove Lemma 1.8 by structural induction on T. As is often the case, we will need to strengthen the inductive hypothesis for the induction to go through. The reason for this is that the read-k decision tree definition only keeps track of the maximum number of times any variable appears in T, whereas we require a more fine-grained accounting of the number of times each variable appears. For a nonempty subset $S \subseteq [n]$, define $m_T(S)$ to be the maximum over $i \in S$ of the number of times x_i appears in the tree T. For example, if T is read-k then $m_T([n]) \leq k$. We will prove the following lemma:

Lemma 2.9. *Let T be a decision tree which computes $f : \{-1,1\}^n \to \{-1,1\}$. Then*

$$\mathbf{Cov}[T] \leq \sum_{S \neq \emptyset} (m_T(S) - 1) \cdot \hat{f}(S)^2.$$

Note that if T is read-k, the right-hand side is at most $(k-1) \cdot \sum_{S \neq \emptyset} \widehat{f}(S)^2 = (k-1) \cdot \mathbf{Var}[f]$, the bound we are looking for.

Proof of Lemma 2.9. We prove this by structural induction on the tree T. The base case we consider is when T queries a single variable.

Base case: In this case, the left and right subtrees are constant functions, so their covariance is zero. For the sum on the right-hand side, any S for which $\widehat{f}(S)$ is nonzero must consist of variables queried by T, in which case $(m_T(S) - 1) \geq 0$. As a result, the right-hand side is always at least 0.

Inductive step: Suppose the root variable of T is x_n. Let T_0 and T_1 be the left and right subtrees of T, respectively. For convenience, we will upper-bound $2 \cdot \mathbf{Cov}[T]$, which can be written as

$$2 \cdot \mathbf{Cov}[T] = 2 \cdot \mathbf{Cov}[g, h] + \mathbf{Cov}[T_0] + \mathbf{Cov}[T_1].$$

We will begin with the first term on the right-hand side. Let J be the set of coordinates which appear in both T_0 and T_1. Because x_n is the root variable, it cannot appear in either T_0 or T_1, so J is a subset of $[n-1]$. Then

$$\begin{aligned} 2 \cdot \mathbf{Cov}[g, h] &= \sum_{S \neq \emptyset} 2 \cdot \widehat{g}(S) \widehat{h}(S) \\ &= \sum_{\emptyset \neq S \subseteq J} 2 \cdot \widehat{g}(S) \widehat{h}(S) \\ &\leq \sum_{\emptyset \neq S \subseteq J} \widehat{g}(S)^2 + \widehat{h}(S)^2, \end{aligned}$$

where the last line holds because $2ab \leq a^2 + b^2$.

Now we focus on the second term. Applying the inductive hypothesis to g and h yields

$$\mathbf{Cov}[T_0] + \mathbf{Cov}[T_1] \leq \sum_{\emptyset \neq S \subseteq [n-1]} (m_{T_0}(S) - 1) \cdot \widehat{g}(S)^2 + (m_{T_1}(S) - 1) \cdot \widehat{h}(S)^2. \tag{4}$$

For any S in the above sum, we have that $m_{T_0}(S) \leq m_T(S)$. This is because T_0 is a subtree of T. However, when $S \subseteq J$ we get the following improved bound: $m_{T_0}(S) \leq m_T(S) - 1$. This holds because every variable in S is queried at least once in T_1, and so it must be queried in T_0 at least one fewer time than in the whole of T. Similarly, all of these inequalities hold when T_0 is replaced with T_1. Rewriting Equation 4,

$$\mathbf{Cov}[T_0] + \mathbf{Cov}[T_1] \leq \sum_{\emptyset \neq S \subseteq J} (m_T(S) - 2) \cdot \left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right) + \sum_{\emptyset \neq S \not\subseteq J} (m_T(S) - 1) \cdot \left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right)$$

Now, if we add $2 \cdot \mathbf{Cov}[g, h]$ to this, we see that it will add 1 to the coefficient of $\widehat{g}(S)^2$ and $\widehat{h}(S)^2$ exactly when $\emptyset \neq S \subseteq J$. As a result,

$$2 \cdot \mathbf{Cov}[g, h] + \mathbf{Cov}[T_0] + \mathbf{Cov}[T_1] \leq \sum_{\emptyset \neq S \subseteq [n-1]} (m_T(S) - 1) \cdot \left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right).$$

The left-hand side is $2 \cdot \mathbf{Cov}[T]$. As for the right-hand side, applying Proposition 2.3 shows that it is equal to

$$2 \cdot \sum_{\emptyset \neq S \subseteq [n-1]} (m_T(S) - 1) \cdot \left(\widehat{f}(S)^2 + \widehat{f}(S \cup \{n\})^2\right).$$

We would be done, except $\widehat{f}(S \cup \{n\})^2$ should have $m_T(S \cup \{n\})$ as its coefficient, not $m_T(S)$. However, $m_T(S) \leq m_T(S \cup \{n\})$ always, so we can perform this replacement. This yields the lemma. □

3 A composition theorem for protocols

In this section we prove Theorem 1.12. Theorem 1.12 concerns several different functions and their spectral distributions defined with respect to different product distributions. We assume here familiarity with Fourier analysis for product distributions over the Boolean cube (see [O'D] for an introduction) and briefly review some basic facts and notation used in the proof.

For a Boolean function $f : \{-1, 1\}^n_\mu \to \{-1, 1\}$, where $\mu = \langle \mu_1, \cdots, \mu_n \rangle$ is a sequence of biases, we think of $\{-1, 1\}^n_\mu$ as endowed with the product distribution that sets each bit independently in $\{-1, 1\}$ with expectation $\mathbf{E}_\mu[x_i] = \mu_i$ and $\mathbf{Var}_\mu[x_i] = 1 - \mu_i^2$. Then the μ-biased Fourier decomposition of f is

$$f = \sum_{S \subseteq [n]} \tilde{f}(S) \phi^\mu_S$$

where

$$\phi^\mu_S(x) = \prod_{i \in S} \frac{x_i - \mu_i}{\mathbf{Var}_\mu[x_i]},$$

and $\tilde{f}(S) = \mathbf{E}_\mu[f \cdot \prod_i i \in S]$. Thus, a spectral sample from $\tilde{h}^2$ is distributed so that each $\mathcal{Y}$ appears with probability $\tilde{h}(\mathcal{Y})^2$.

Now we proceed to prove Theorem 1.12. Let P_f be a C-good protocol for f under η and $P_1, \ldots, P_k$ be C-good protocols for $g_1, \ldots, g_k$ under μ. Recall that these protocols are prefix-free. Now, consider a spectral sample $\mathcal{Y} \sim \tilde{h}^2$ and the following protocol $P_h(\mathcal{Y})$:

1. Let $S \subseteq [k]$ be the set containing those $i \in [k]$ such that $\mathcal{Y}_i \neq \emptyset$.
2. Output $P_f(S)$.
3. For each $i \in S$, output $P_i(\mathcal{Y}_i)$.

Because the subprotocols are prefix-free, $P_h(\mathcal{Y})$ is a prefix-free encoding of $\mathcal{Y}$. This is because if one scans the output of $P_h(\mathcal{Y})$ from left-to-right, the first prefix which could be output by $P_f(\cdot)$ must actually be the output of $P_f(S)$. This gives a description of the set S, from which one can recover $\mathcal{Y}_1, \ldots, \mathcal{Y}_k$ by a similar process. We will show that if P_f and $P_1, \ldots, P_k$ are efficient, then P_h is efficient as well. To begin, we will need the following pair of claims:

Claim 3.1. *Conditioned on $\mathcal{Y}_i \neq \emptyset$, $\mathcal{Y}_i$ is distributed as $\widetilde{g_i}^2 \setminus \emptyset$.*

Claim 3.2. *The set S is distributed as $\widetilde{f}^2 \setminus \emptyset$.*

The proofs of these claims, as well as the basic Fourier analytic facts used to prove them, may be found in the Appendix D. We now prove the composition theorem for C-good protocols.

Lemma 3.3. *If P_i is a C-good protocol for each g_i and P_f is a C-good protocol for f, then P_h is a C-good protocol for h.*

Proof. The expected output size of the protocol is

$$\begin{aligned}\mathbf{E}[|P_h(\mathcal{Y})|] &= \mathbf{E}\left[|P_f(S)| + \sum_{i \in S} |P_i(\mathcal{Y}_i)|\right] \\ &= \mathbf{E}\left[|P_f(S)| + \sum_{i=1}^{k} \mathbf{1}[\mathcal{Y}_i \neq \emptyset] \cdot |P_i(\mathcal{Y}_i)|\right].\end{aligned}$$

First, we upper bound the second term in the expectation. For a fixed i,

$$\mathbf{E}\left[\mathbf{1}[\mathcal{Y}_i \neq \emptyset] \cdot |P_i(\mathcal{Y}_i)|\right] = \mathbf{Pr}[\mathcal{Y}_i \neq \emptyset] \cdot \mathbf{E}\left[|P_i(\mathcal{Y}_i)| \middle| \mathcal{Y}_i \neq \emptyset\right]. \tag{5}$$

From Claim 3.1, $\mathcal{Y}_i$ conditioned on $\mathcal{Y}_i \neq \emptyset$ is distributed as $\widetilde{g_i}^2 \setminus \emptyset$. Thus, as P_i is a C-good protocol for g_i, we may upper bound $\mathbf{E}\left[|P_i(\mathcal{Y}_i)| \middle| \mathcal{Y}_i \neq \emptyset\right]$ with the expression in the definition of a C-good protocol, except where that definition uses an $\mathcal{Y}$, we have instead $\mathcal{Y}_i \middle| (\mathcal{Y}_i \neq \emptyset)$. Note that $\mathbf{Pr}[\mathcal{Y}_i \neq \emptyset] \cdot \mathbf{E}\left[|\mathcal{Y}_i| \middle| \mathcal{Y}_i \neq \emptyset\right] = \mathbf{E}[|\mathcal{Y}_i|]$ and that $\mathbf{Pr}[\mathcal{Y}_i \neq \emptyset] \cdot \mathbf{Pr}[j \in \mathcal{Y}_i \middle| \mathcal{Y}_i \neq \emptyset] = \mathbf{Pr}[j \in \mathcal{Y}_i]$. As a result, the upper bound we get on Equation 5 is

$$C \cdot (\mathbf{E}[|\mathcal{Y}_i|] - \mathbf{Pr}[\mathcal{Y}_i \neq \emptyset]) + \sum_j \mathbf{Pr}[j \in \mathcal{Y}_i] \cdot \log \frac{1}{\mathbf{Var}_\mu[x_j]} + \mathbf{Pr}[\mathcal{Y}_i \neq \emptyset] \cdot \log \mathop{\mathbf{Var}}_{\mu}[g_i].$$

Note that $\mathcal{Y}_i \neq \emptyset$ exactly when $i \in S$. As a result, summing this over all $i \in [k]$ yields

$$\begin{aligned}\mathbf{E}\left[\sum_{i \in S} |P_i(\mathcal{Y}_i)|\right] \leq\ & C \cdot (\mathbf{E}[|\mathcal{Y}|] - \mathbf{E}[|S|]) \\ & + \sum_{j \in [n]} \mathbf{Pr}[j \in \mathcal{Y}] \cdot \log \frac{1}{\mathbf{Var}_\mu[x_j]} + \sum_{i \in [k]} \mathbf{Pr}[i \in S] \cdot \log \mathop{\mathbf{Var}}_{\mu}[g_i].\end{aligned} \tag{6}$$

For the first term in the expectation, we know by Claim 3.2 that the random variable S defined in the protocol is distributed according to $\widetilde{f}^2 \setminus \emptyset$. Thus, because P_f is a C-good protocol,

$$\mathbf{E}[P_f(S)] \leq C \cdot (\mathbf{E}[|S|] - 1) + \sum_{i \in [k]} \mathbf{Pr}[i \in S] \cdot \log \frac{1}{\mathbf{Var}_\eta[y_i]} + \log \mathop{\mathbf{Var}}_\eta[f].$$

Note that $\mathbf{Var}_\eta[y_i] = \mathbf{Var}_\mu[g_i]$ and $\mathbf{Var}_\eta[f] = \mathbf{Var}_\mu[h]$. As a result, adding these together yields

$$\mathbf{E}[|P(\mathcal{Y})|] \leq C \cdot (\mathbf{E}[|\mathcal{Y}|] - 1) + \sum_{j \in [n]} \mathbf{Pr}[j \in \mathcal{Y}] \cdot \log \frac{1}{\mathbf{Var}_\mu[x_j]} + \log \mathop{\mathbf{Var}}_\mu[h],$$

which yields the theorem. □

References

[CKLS13] Sourav Chakraborty, Raghav Kulkarni, Satyanarayana Lokam, and Nitin Saurabh. Upper bounds on Fourier Entropy. Electronic Colloquium on Computational Complexity TR13-052, 2013. 1, 1

[FK96] Ehud Friedgut and Gil Kalai. Every monotone graph property has a sharp threshold. *Proceedings of the American Mathematical Society*, 124(10):2993–3002, 1996. (document), 1

[GKK08] Parikshit Gopalan, Adam Kalai, and Adam Klivans. Agnostically learning decision trees. In *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, pages 527–536, 2008. 1

[Kal07] Gil Kalai. The entropy/influence conjecture. Posted on Terence Tao's *What's new* blog, http://terrytao.wordpress.com/2007/08/16/gil-kalai-the-entropyinfluence-conjecture/, 2007. 1

[KKL88] Jeff Kahn, Gil Kalai, and Nathan Linial. The influence of variables on Boolean functions. In *Proceedings of the 29th Annual IEEE Symposium on Foundations of Computer Science*, pages 68–80, 1988. 1

[KLW10] Adam Klivans, Homin Lee, and Andrew Wan. Mansour's Conjecture is true for random DNF formulas. In *Proceedings of the 23rd Annual Conference on Learning Theory*, pages 368–380, 2010. 1

[KMS12] Nathan Keller, Elchanan Mossel, and Tomer Schlank. A note on the Entropy/Influence conjecture. *Discrete Mathematics*, 312(22):3364–3372, 2012. 1.1

[Man94] Yishay Mansour. Learning Boolean functions via the Fourier Transform. In Vwani Roychowdhury, Kai-Yeung Siu, and Alon Orlitsky, editors, *Theoretical Advances in Neural Computation and Learning*, chapter 11, pages 391–424. Kluwer Academic Publishers, 1994. 1

[O'D] Ryan O'Donnell. *Analysis of Boolean Functions*. 2.1, 3

[O'D03] Ryan O'Donnell. *Computational applications of noise sensitivity*. PhD thesis, Massachusetts Institute of Technology, 2003. 1

[OT13] Ryan O'Donnell and Li-Yang Tan. A composition theorem for the Fourier Entropy-Influence conjecture. In *Proceedings of the 40th Annual International Colloquium on Automata, Languages and Programming*, pages 780–791, 2013. (document), 1, 1, 2, 1.3, 1.3, 1.3, C

[OWZ11] Ryan O'Donnell, John Wright, and Yuan Zhou. The Fourier Entropy-Influence conjecture for certain classes of Boolean functions. In *Proceedings of the 38th Annual International Colloquium on Automata, Languages and Programming*, pages 330–341, 2011. 1, 1.1, 2, 2.1, B, B

[Sha48] Claude E. Shannon. A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423, 623–656, July, October 1948. 1.4

A Decision tree proofs

We will repeatedly use the following proposition, which relates the Fourier coefficients of f to the Fourier coefficients of its subfunctions g and h.

Proposition A.1. *Let f be computed by a decision tree T whose left and right subfunctions are g and h, respectively. If x_i is at the root of T and S is any subset of $[n] \setminus \{i\}$, then*

$$\widehat{f}(S)^2 + \widehat{f}(S \cup \{i\})^2 = \frac{1}{2}\left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right).$$

Proof. Write f as

$$f = \left(\frac{1+x_i}{2}\right) g + \left(\frac{1-x_i}{2}\right) h.$$

For any $S \subseteq [n] \setminus \{i\}$, $\widehat{f}(S) = \frac{1}{2}(\widehat{g}(S) + \widehat{h}(S))$ and $\widehat{f}(S \cup \{i\}) = \frac{1}{2}(\widehat{g}(S) - \widehat{h}(S))$. As a result,

$$\widehat{f}(S)^2 + \widehat{f}(S \cup \{i\})^2 = \frac{1}{2}\left(\widehat{g}(S)^2 + \widehat{h}(S)^2\right). \qquad \square$$

We will also use the following proposition, which relates the influences of f to the influences of its subfunctions.

Proposition A.2. *Assume the setup from Proposition 2.3. Then for a coordinate $j \neq i$,*

$$\mathbf{Inf}_j[f] = \frac{1}{2} \cdot (\mathbf{Inf}_j[g] + \mathbf{Inf}_j[h]).$$

Proof.

$$\begin{aligned}\mathbf{Inf}_j[f] &= \mathbf{Pr}[f(\boldsymbol{x}) \neq f(\boldsymbol{x}^{\oplus j})] \\ &= \frac{1}{2}\mathbf{Pr}[g(\boldsymbol{x}) \neq g(\boldsymbol{x}^{\oplus j})] + \frac{1}{2}\mathbf{Pr}[h(\boldsymbol{x}) \neq h(\boldsymbol{x}^{\oplus j})] = \frac{1}{2}\left(\mathbf{Inf}_j[g] + \mathbf{Inf}_j[h]\right).\end{aligned} \qquad \square$$

B Proof of Lemma 1.5

In this section, we give a proof of Lemma 1.5, which was implicit in [OWZ11]; the proof we give here, included for completeness, is essentially the same. First, we have the following lemma:

Lemma B.1. *Let $f : \{-1,1\}^n \to \{-1,1\}$ and write $\hat{f}(\emptyset)^2 = 1 - \epsilon$. Then $2 \cdot \mathbf{Inf}[f] \geq h(\epsilon)$, where $h(\cdot)$ is the binary entropy function.*

Proof. First, we may assume $\epsilon \neq 0, 1$, otherwise the result is trivial. Now, $(1-\epsilon)\log\frac{1}{1-\epsilon} \leq \frac{1}{\ln 2}\epsilon \leq 2\epsilon$, so

$$h(\epsilon) = \epsilon \log \frac{1}{\epsilon} + (1-\epsilon)\log\frac{1}{1-\epsilon} \leq \epsilon \log\frac{1}{\epsilon} + 2\epsilon.$$

By Proposition 2 of [OWZ11], the right-hand side is at most $2 \cdot \mathbf{Inf}[f]$, and the lemma follows. □

Lemma B.2 (Restatement of Lemma 1.5)**.** *Suppose there is an almost prefix-free protocol for $\mathcal{X}$ with length B and alphabet Σ. Then $\mathbf{H}[\mathcal{X}] \leq \log_2|\Sigma| \cdot B + 2 \cdot \mathbf{Inf}[f]$.*

Proof. Write $\hat{f}(\emptyset)^2 = 1 - \epsilon$. If $\epsilon = 0$ then $\mathbf{H}[\mathcal{X}] = 0$, so the lemma follows. Otherwise, let $\mathcal{Y}$ be the indicator that $\mathcal{X} = \emptyset$. Then $\mathbf{H}[\mathcal{X}|\mathcal{Y}=0] \leq \log_2|\Sigma|\ B/\epsilon$ by the source coding theorem, as the protocol outputs B/ϵ characters on average conditioned on $\mathcal{X}$ being nonempty.

$$\begin{aligned}
\mathbf{H}[\mathcal{X}] &= \mathbf{H}[\mathcal{X}, \mathcal{Y}] \\
&= \mathbf{H}[\mathcal{Y}] + \mathbf{H}[\mathcal{X}|\mathcal{Y}] && \text{(conditional entropy)} \\
&= \mathbf{H}[\mathcal{Y}] + (1-\epsilon)\cdot\mathbf{H}[\mathcal{X}|\mathcal{Y}=1] + \epsilon\cdot\mathbf{H}[\mathcal{X}|\mathcal{Y}=0] \\
&\leq \mathbf{H}[\mathcal{Y}] + \log_2|\Sigma|\cdot B && \text{(using } \mathbf{H}[\mathcal{X}|\mathcal{Y}=1]=0)
\end{aligned}$$

Because $\mathcal{Y}$ is a $(1-\epsilon)$-biased random bit, $\mathbf{H}[\mathcal{Y}] = h(\epsilon)$, where $h(\cdot)$ is the binary entropy function. Thus, we may apply Lemma B.1 and get that $\mathbf{H}[\mathcal{X}] \leq 2 \cdot \mathbf{Inf}[f] + \log_2|\Sigma| \cdot B$. □

C Parallelizing the Protocol

The performance of Shannon's code gives the following guarantee:

Fact C.1. *Let $\mathcal{X}^1, \ldots, \mathcal{X}^t$ be t independent samples drawn from $\widetilde{f^2} \setminus \emptyset$. Then there is a prefix-free protocol P_f^t for which*

$$\mathbf{E}\left[|P_f^t(\mathcal{X}^1, \ldots, \mathcal{X}^t)|\right] \leq t \cdot \boldsymbol{H}\left[\widetilde{f^2} \setminus \emptyset\right] + 1.$$

In other words, the average number of bits used per copy of $\mathcal{X}$ is $1/t$ more than the theoretical best. In the limit as t tends to ∞, the excess number of bits tends to 0. Using this, we will show that the protocol from Section 3 may be analyzed as if the subprotocols are optimally efficient. We will do this by showing an efficient protocol to communicate sets $\mathcal{Y}^1, \ldots, \mathcal{Y}^t \sim \widetilde{h^2} \setminus \emptyset$ which are chosen independently. As before, use $\mathcal{Y}_i^j$ to denote the restriction of $\mathcal{Y}^j$ to the coordinates relevant to g_i. We will assume that we have the efficient protocol P_f^t guaranteed by Fact C.1. In addition, for each $i \in [k]$ and $m \in [t]$ we will use the protocol P_i^m which Fact C.1 guarantees will efficiently communicate m samples from $\widetilde{g_i^2} \setminus \emptyset$. Now, consider the following protocol $P_h^t(\mathcal{Y}^1, \ldots, \mathcal{Y}^t)$:

1. For each $j \in [t]$, let $S^j \subseteq [k]$ be the set containing those i such that $\mathcal{Y}_i^j \neq \emptyset$.
2. Output $P_f^t(S^1, \ldots, S^t)$.
3. For each $i \in [k]$:
   (a) Let $j_1, \ldots, j_m$ be the indices of the nonempty $\mathcal{Y}_i^j$ (in order).
   (b) If $m \neq 0$, output $P_i^m(\mathcal{Y}_i^{j_1}, \ldots, \mathcal{Y}_i^{j_m})$. Otherwise, output nothing.

The following lemma, which may be compared to Proposition 3.2 in [OT13], gives the performance of this protocol and suffices to recover their composition theorem for entropy.

Lemma C.2. *Let S be distributed as in the protocol from Section 3. In the limit as $t \to \infty$,*

$$\frac{1}{t} \cdot \mathbf{E}\left[|P_h^t(\mathcal{Y}^1, \ldots, \mathcal{Y}^t)|\right] = \boldsymbol{H}\left[\widetilde{f}^2 \setminus \emptyset\right] + \sum_{i \in [k]} \mathbf{Pr}[i \in S] \cdot \boldsymbol{H}\left[\widetilde{g_i}^2 \setminus \emptyset\right].$$

Proof sketch. Fix a coordinate $i \in [k]$ and consider the number m of nonempty $\mathcal{Y}_i^j$s. From Claim 3.1, we know that if $\mathcal{Y}_i^j$ is nonempty, then it is distributed as $\widetilde{g_i}^2 \setminus \emptyset$. As a result, for a fixed value of m, Fact C.1 tells us that the expected number of bits that P_i^m outputs per $\mathcal{Y}_i^j$ is at most $1/m$ in excess of $\mathbf{H}\left[\widetilde{g_i}^2 \setminus \emptyset\right]$. Now, m is distributed as $\mathrm{Bin}(t, r)$, where r is a probability independent of t. Thus, by taking $t \to \infty$ the expectation of $1/m$ (when m is nonzero) will tend towards 0. As a result, we may assume that $\mathbf{H}\left[\widetilde{g_i}^2 \setminus \emptyset\right]$ bits are used in expectation to communicate each nonzero $\mathcal{Y}_i^j$. A similar argument shows that we may assume that $\mathbf{H}[\widetilde{f}^2 \setminus \emptyset]$ bits are used in expectation to communicate each S^j.

Aside from packaging the different sets together when calling the subprotocols, the protocol acts as t independent copies of the protocol from Section 3. Let us focus on the case when $j = 1$. Then the expected number of bits spent outputting the sets for which $j = 1$ is

$$\mathbf{H}\left[\widetilde{f}^2 \setminus \emptyset\right] + \sum_{i \in [k]} \mathbf{Pr}[i \in S^1] \cdot \mathbf{H}\left[\widetilde{g_i}^2 \setminus \emptyset\right].$$

As S^1 is distributed identically to S in the protocol from Section 3, we may replace the event $i \in S^1$ with $i \in S$. Averaging this over all $j \in [t]$ yields the lemma. □

D Proofs of Claims

First, we recall several basic facts regarding μ-biased Fourier analysis. For $S \neq T$ and $S \neq \emptyset$, we have $\mathbf{E}_\mu[\phi_S^\mu] = 0$ and $\mathbf{E}_\mu[\phi_S^\mu \cdot \phi_T^\mu] = 0$. We also have Parseval's inequality, which states that for $f : \{-1, 1\}^n \to \mathbb{R}$, the equality $\sum_{S \subseteq [n]} \tilde{f}(S)^2 = \mathbf{E}_\mu[f^2]$ holds.

We now prove the following proposition, from which the claims follow immediately.

Proposition D.1. *Given the setup of the first protocol,*

$$\widetilde{h}(\mathcal{Y}) = \widetilde{f}(S) \prod_{i \in S} \frac{\widetilde{g}(\mathcal{Y}_i)}{\sigma_i}.$$

Proof. Let $\eta_i = \mathbf{E}_\mu[g_i]$ and $\sigma_i^2 = \mathbf{Var}_\mu[g_i]$. Let S be as defined in the protocol. Then

$$\begin{aligned}
\widetilde{h}(\mathcal{Y}) &= \mathop{\mathbf{E}}_{\boldsymbol{x} \sim \mu}[h(\boldsymbol{x}) \cdot \phi_{\mathcal{Y}}^\mu(\boldsymbol{x})] \\
&= \mathop{\mathbf{E}}_{\boldsymbol{x}}\left[f(g_1(\boldsymbol{x}), \ldots, g_k(\boldsymbol{x})) \cdot \prod_{j \in S} \phi_{\mathcal{Y}_j}^\mu(\boldsymbol{x})\right] \\
&= \sum_{T \subseteq [k]} \widetilde{f}(T) \mathop{\mathbf{E}}_{\boldsymbol{x}}\left[\phi_T^\eta(g_1(\boldsymbol{x}), \ldots, g_k(\boldsymbol{x})) \cdot \prod_{j \in S} \phi_{\mathcal{Y}_j}^\mu(\boldsymbol{x})\right] \\
&= \sum_{T \subseteq [k]} \widetilde{f}(T) \mathop{\mathbf{E}}_{\boldsymbol{x}}\left[\prod_{i \in T}\left(\frac{g_i(\boldsymbol{x}) - \eta_i}{\sigma_i}\right) \prod_{j \in S} \phi_{\mathcal{Y}_j}^\mu(\boldsymbol{x})\right].
\end{aligned}$$

A standard calculation shows that the expectation is nonzero only if $S = T$. In this case, the expectation is equal to

$$\prod_{i \in S} \mathop{\mathbf{E}}_{\boldsymbol{x}} \left[\left(\frac{g_i(\boldsymbol{x}) - \eta_i}{\sigma_i} \right) \cdot \phi^{\mu}_{\mathcal{Y}_i}(\boldsymbol{x}) \right] = \prod_{i \in S} \frac{\widetilde{g}(\mathcal{Y}_i)}{\sigma_i},$$

where the equality holds because $\mathcal{Y}_i$ is nonempty, so the shift by η_i doesn't affect the calculation. The proposition now follows. □

Now we prove the claims:

Proof of Claim 3.1. Condition $\mathcal{Y}$ on $\mathcal{Y}_i \neq \emptyset$ and on any values for $\mathcal{Y}_1, \ldots, \mathcal{Y}_{i-1}, \mathcal{Y}_{i+1}, \ldots, \mathcal{Y}_k$. Then by Proposition D.1, $\mathcal{Y}_i$ is distributed as $\widetilde{g_i}^2 \setminus \emptyset$. As this holds conditioned on any values for the $\mathcal{Y}_j$'s, $j \neq i$, this also holds conditioned only on $\mathcal{Y}_i \neq \emptyset$. □

Proof of Claim 3.2. First, because f and h have the same mean, they also have the same variance, i.e.

$$\sum_{\mathcal{Y} \neq \emptyset} \widetilde{h}^2(\mathcal{Y}) = \sum_{S \neq \emptyset} \widetilde{f}^2(S).$$

Next, fix a particular value of $S \subseteq [k]$, $S \neq \emptyset$. The sets $\mathcal{Y}$ for which the protocol selects this particular S are those for which $\mathcal{Y}_i \neq \emptyset \iff i \in S$. Then the probability S is selected is just the sum over these sets:

$$\begin{aligned}
\sum_{\mathcal{Y}: \mathcal{Y}_i \neq \emptyset \iff i \in S} \widetilde{h}(\mathcal{Y})^2 &= \sum_{\mathcal{Y}: \mathcal{Y}_i \neq \emptyset \iff i \in S} \widetilde{f}(S)^2 \prod_{i \in S} \frac{\widetilde{g}(\mathcal{Y}_i)^2}{\sigma_i^2} && \text{(by Proposition D.1)} \\
&= \widetilde{f}(S)^2 \prod_{i \in S} \sum_{\mathcal{Y}_i \neq \emptyset} \frac{\widetilde{g}(\mathcal{Y}_i)^2}{\sigma_i^2} \\
&= \widetilde{f}(S)^2.
\end{aligned}$$

Combining these two facts yields the claim. □

E Small influence counterexample

Suppose we could prove Theorem 1.2 without the restriction on the function's total influence, i.e. the following statement:

Conjecture E.1. Suppose $f : \{-1,1\}^n \to \{-1,1\}$ is computable by a decision tree with expected depth d, and let $\mathcal{X} \sim \widehat{f}^2$. Then $\mathbf{H}[\mathcal{X}] \leq C \cdot d \cdot \mathbf{Inf}[f]$, for some absolute constant C.

This appears to be a weaker conjecture than the FEI conjecture. However, in this section we will show that this statement implies the FEI conjecture, at least for functions with sufficiently large influence.

Proposition E.2. *Suppose Conjecture E.1 were true for some constant C. Let $f : \{-1,1\}^n \to \{-1,1\}$, and let $\mathcal{X} \sim \widehat{f}^2$. If $\mathbf{Inf}[f] \geq \log(n)$, then $\mathbf{H}[\mathcal{X}] \leq C' \cdot \mathbf{Inf}[f]$, where C' is some other absolute constant.*

Although this only shows that Conjecture E.1 implies a restricted form of the FEI conjecture, this restricted form does not appear to be especially easier than the full FEI conjecture. Thus, the restriction in Theorem 1.2 that f have large influence is a natural one.

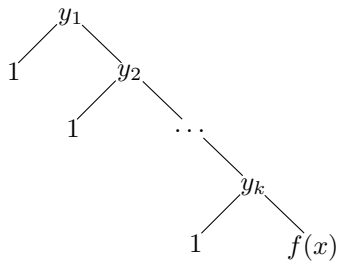


Figure 3: A decision tree computing $g(x, y)$.

Let $f : \{-1, 1\}^n \to \{-1, 1\}$ have $\mathbf{Inf}[f] \geq \log n$. We prove Proposition E.2 by "hiding" f in a low expected-depth decision tree. The resulting decision tree still has low expected-depth, and its spectral entropy and total influence terms are roughly proportional to f's. Thus, applying Conjecture E.1 to the decision tree shows that f itself satisfies the FEI conjecture.

Proof. Let $f : \{-1, 1\}^n \to \{-1, 1\}$ have $\mathbf{Inf}[f] \geq \log n$, and let $\mathcal{X} \sim \widehat{f}^2$. We begin with the assumption that f is balanced, i.e. that $\mathbf{E}[f] = 0$, and we will later reduce the general case to this case. For simplicity, assume that n is a power of two. Consider the new function $g(x, y)$ defined as

$$g(x, y) = \begin{cases} f(x) & \text{if } AND(y_1, \ldots, y_k) = -1, \\ 1 & otherwise. \end{cases}$$

Pictorially, refer to Figure 3, where $f(x)$ is computed by some decision tree. Since f can be trivially computed by a decision tree of depth n, the decision tree pictured computes f with expected depth at most $2 + n/2^k$. By choosing $k = \log_2 n$, then this decision tree is depth 3.

Each variable y_i is influential only when the rest of the y_j's are -1 and $f(x) = -1$ (which happens half of the time because f is balanced), so the influence of each y_i is exactly $1/2^k$. Each of the x_i variables is influential only when all of the $y_i's$ are -1, so the influence of variable x_i on g is exactly $\mathbf{Inf}_i[f]/2^k$. As a result,

$$\mathbf{Inf}[g] = \frac{k + \mathbf{Inf}[f]}{2^k} = \frac{\log n + \mathbf{Inf}[f]}{n}.$$

To compute the entropy, we can first write g as

$$g(x, y) = 1 - \frac{1}{2^k} \sum_S \chi_S(y) \left(\frac{1}{2} + \frac{f(x)}{2} \right).$$

From this, we can easily read off some of the Fourier coefficients of g: if $S \subseteq [k]$ and $T \subseteq [n]$ are both nonempty, then $\widehat{g}(S, T) = -\widehat{f}(T)/2^{k+1}$. As a result, if $\mathcal{X}' \sim \widehat{g}^2$, then we can lower bound $\mathbf{H}[\mathcal{X}']$ by summing over the terms in the entropy formula corresponding to these subsets:

$$\begin{aligned} \mathbf{H}[\mathcal{X}'] &\geq \sum_{S,T \neq \emptyset} \frac{\widehat{f}(T)^2}{2^{2k+2}} \log \left(\frac{2^{2k+2}}{\widehat{f}(T)^2} \right) \\ &\geq \sum_T \frac{\widehat{f}(T)^2}{2^{k+3}} \log \left(\frac{2^{2k+2}}{\widehat{f}(T)^2} \right) \\ &\geq \sum_T \frac{(2k+2) \cdot \widehat{f}(T)^2}{2^{k+3}} + \sum_T \frac{\widehat{f}(T)^2}{2^{k+3}} \log \left(\frac{1}{\widehat{f}(T)^2} \right) \\ &= \frac{2k + 2 + \mathbf{H}[\mathcal{X}]}{2^{k+3}} = \frac{2 \log n + 2 + \mathbf{H}[\mathcal{X}]}{8n}. \end{aligned}$$

Here the second inequality follows because the sum is over $2^k - 1 \geq 2^{k-1}$ sets S and because $\widehat{f}(\emptyset) = 0$. The second-to-last equality follows because f is mean-zero, so $\sum_T \widehat{f}(T)^2 = 1$.

Now, applying Conjecture E.1 to g, we have that

$$\frac{3\log n + \mathbf{H}[\mathcal{X}]}{8n} \leq \mathbf{H}[\mathcal{X}'] \leq C \cdot 3 \cdot \mathbf{Inf}[g] = C \cdot 3 \cdot \frac{\log n + \mathbf{Inf}[f]}{n}.$$

This can be rearranged as

$$\mathbf{H}[\mathcal{X}] \leq (24C - 3) \cdot \log n + 24C \cdot \mathbf{Inf}[f].$$

Thus, if $\mathbf{Inf}[f] \geq \log n$, then $\mathbf{H}[\mathcal{X}] \leq C'\mathbf{Inf}[f]$, where $C' = 48C - 3$.

Now, if f is not balanced, consider the function $g(x_1, \ldots, x_n, x_{n+1}) = x_{n+1} \cdot f(x)$. Then g is balanced, has the same Fourier entropy as f, and $\mathbf{Inf}[g] = \mathbf{Inf}[f] + 1$. As we have just shown,

$$\begin{aligned}
\mathbf{H}[\mathcal{X}] &= \mathbf{H}[\widehat{g}^2] \\
&\leq C' \cdot \mathbf{Inf}[g] \\
&= C' \cdot (\mathbf{Inf}[f] + 1) \\
&\leq C' \cdot (\mathbf{Inf}[f] + \log n) \\
&\leq (C' + 1) \cdot \mathbf{Inf}[f].
\end{aligned}$$

Here, the last inequality uses the fact that $\log n \leq \mathbf{Inf}[f]$. Thus, f satisfies the FEI conjecture with constant $C' + 1$. □