A Brief Introduction to Fourier Analysis on the Boolean Cube

Ronald de Wolf[*]

Abstract

We give a brief introduction to the basic notions of Fourier analysis on the Boolean cube, illustrated and motivated by a number of applications in theoretical computer science.

1 Introduction

Fourier transforms are widely used in mathematics, computer science, and engineering. Examples include signal processing, data compression, fast multiplication of polynomials, quantum computing, as well as many others. The Fourier transform most commonly used is the one over cyclic groups. Here one decomposes a signal or function as a sum of periodic functions such as $\chi_y(x) = e^{2\pi ixy/n}$ or (in the real case) sines and cosines.

In the study of functions of n Boolean variables, however, the most natural Fourier transform to consider is the one over the Abelian group $\mathbb{Z}_2^n$. This is known as Fourier analysis over the Boolean cube, and has over the past two decades become one of the most important and versatile tools for theoretical computer scientists. The main purpose of this paper is to give a first introduction to the basic notions and properties of this area (Section 2) and to illustrate and motivate them by means of a few relatively simple but elegant applications from diverse areas (Sections 3 and 4). The intended audience is theoretical computer scientists interested in adding these techniques to their toolbox. The selection of applications is somewhat biased by the author's own experience in learning this material in recent years, and is by no means a complete survey—such a survey would probably require a whole book by now. However, in Section 5 we give pointers for further reading.

2 Definitions and basic properties

2.1 The vector space of functions on the Boolean cube

Consider the 2^n-dimensional vector space of all functions $f : \{0,1\}^n \to \mathbb{R}$. We define an inner product on this space by

$$\langle f, g\rangle = \frac{1}{2^n}\sum_{x\in\{0,1\}^n} f(x)g(x) = \mathbb{E}[f\cdot g],$$

where the latter expectation is taken uniformly over all $x \in \{0,1\}^n$. This defines the ℓ_2-norm

$$||f||_2 = \sqrt{\langle f, f\rangle} = \sqrt{\mathbb{E}[f^2]}.$$

[*] CWI Amsterdam. Email rdewolf@cwi.nl. Partially supported by a Veni grant from the Netherlands Organization for Scientific Research (NWO), and by the European Commission under the Integrated Project Qubit Applications (QAP) funded by the IST directorate as Contract Number 015848.

2.2 The Fourier transform

It will be convenient to identify a set $S \subseteq [n] = \{1, \ldots, n\}$ with its characteristic vector $S \in \{0,1\}^n$. For example for $S = \{1,3\} \subseteq [3]$ we can also write $S = (1,0,1)$ or $S = 101$. We will often go back and forth between these two. For each $S \subseteq [n]$, define a function $\chi_S : \{0,1\}^n \to \{\pm 1\}$ by

$$\chi_S(x) = (-1)^{S \cdot x},$$

where $S \cdot x = \sum_{i=1}^n S_i x_i = \sum_{i \in S} x_i$. It is easy to see that

$$\langle \chi_S, \chi_T \rangle = \delta_{ST} = \left\{ \begin{array}{ll} 1 & \text{if } S = T \\ 0 & \text{if } S \neq T \end{array} \right. ,$$

hence the set of all χ_S is an orthonormal basis (called the *Fourier basis*) for the space of all real-valued functions on $\{0,1\}^n$. Of course, there are many different bases for this space. What makes the Fourier basis particularly useful for computer science is that the basis functions themselves have a simple computational interpretation, namely as parity functions: $\chi_S(x) = -1$ if the number of S-variables having value 1 in the input x is odd, and $\chi_S(x) = 1$ if that number is even.

For any $f : \{0,1\}^n \to \mathbb{R}$ we can define another function $\widehat{f} : \{0,1\}^n \to \mathbb{R}$ by

$$\widehat{f}(S) = \langle f, \chi_S \rangle = \mathbb{E}[f \cdot \chi_S].$$

The linear map

$$\mathcal{F} : f \mapsto \widehat{f}$$

is called the *Fourier transform*, $\widehat{f} = \mathcal{F}(f)$ is the Fourier transform *of* f, and $\widehat{f}(S)$ is the Fourier *coefficient* of f at S. This $\widehat{f}(S)$ may be viewed as a measure of the correlation between f and χ_S. The set of Fourier coefficients is also called the *Fourier spectrum* of f. Since the χ_S form an orthonormal basis, the relation between f and $\widehat{f}$ is

$$f = \sum_S \widehat{f}(S) \chi_S. \tag{1}$$

Note that $f(\emptyset) = \sum_S \widehat{f}(S)$ is the sum of all Fourier coefficients, while $\widehat{f}(\emptyset) = \mathbb{E}[f]$ is the average function value. A function is constant if, and only if, $\widehat{f}(S) = 0$ for all $S \neq \emptyset$.

The *degree* of f is $\deg(f) = \max\{|S| \mid \widehat{f}(S) \neq 0\}$. In particular the degree of the basis function χ_S is $|S|$, the number of variables it depends on. Since $(-1)^{x_i} = 1 - 2x_i$, the Fourier expansion Eq. (1) represents f as an n-variate *polynomial* over the real numbers, of degree $\deg(f)$.

Let us consider some simple examples. If $f = \chi_S$ then $\widehat{f}(S) = 1$ while all other Fourier coefficients are 0. If $f(x) = \sum_{i \in S} x_i \bmod 2$ is a parity function in the usual 0/1 notation, then $\widehat{f}(\emptyset) = 1/2$ and $\widehat{f}(S) = -1/2$; all other coefficients are 0. The special case where $S = \{i\}$ (i.e., $f(x) = x_i$) is known as a *dictator* function, since its value is determined by only one variable. A *k-junta* is a function depending on at most k variables; equivalently, there is a set $J \subseteq [n]$ of size k such that $\widehat{f}(S) = 0$ whenever $S \not\subseteq J$. Finally, if we pick a function $f : \{0,1\}^n \to \{\pm 1\}$ uniformly at random, then each Fourier coefficient is normally distributed with mean 0 and variance $1/2^n$.

Because the χ_S form an orthonormal basis, we immediately get the following equality:

$$\langle f, g \rangle = \sum_{S,T} \widehat{f}(S) \widehat{g}(T) \langle \chi_S, \chi_T \rangle = \sum_S \widehat{f}(S) \widehat{g}(S).$$

In particular, with $f = g$ we obtain *Parseval's Identity*:

$$||f||_2^2 = \sum_S \widehat{f}(S)^2.$$

This also implies

$$||f - g||_2^2 = \sum_S (\widehat{f}(S) - \widehat{g}(S))^2.$$

As an example, suppose f is a probability distribution. Then we can analyze the ℓ_2-distance between f and the uniform distribution $g(x) = 1/2^n$ as follows:

$$||f - g||_2^2 = \sum_S (\widehat{f}(S) - \widehat{g}(S))^2 = \sum_{S \neq \emptyset} \widehat{f}(S)^2,$$

where we used $\widehat{f}(\emptyset) = \widehat{g}(\emptyset) = 1/2^n$, and $\widehat{g}(S) = 0$ whenever $S \neq \emptyset$.

Notational variants. The definition of $\mathcal{F}$ used here has become the standard one in computer science, though one occasionally sees different normalizations. If we defined the inner product with a $2^{-n/2}$ instead of 2^{-n}, then $\mathcal{F}$ would be its own inverse and Parseval would simply state $||f||_2 = ||\widehat{f}||_2$. However, with this modified inner product the χ_S functions no longer have norm 1.

Another variation is to view the variables as ± 1-valued instead of $0/1$-valued and to consider functions on $\{\pm 1\}^n$. In this case the function χ_S is simply the product of the S-variables, and the Fourier representation is simply an n-variate multilinear polynomial over the reals, with $\widehat{f}(S)$ as the coefficient of the monomial χ_S. Similarly, depending on what is more convenient, we can treat the value of a Boolean function as $0/1$-valued or as ± 1-valued. An advantage of the latter is that $\sum_S \widehat{f}(S)^2 = \mathbb{E}[f^2] = 1$ (by Parseval), which allows us to treat the squared Fourier coefficients as probabilities.

2.3 Convolution

Given any two functions $f, g : \{0,1\}^n \to \mathbb{R}$, we define their *convolution* $f * g : \{0,1\}^n \to \mathbb{R}$ by

$$(f * g)(x) = \frac{1}{2^n} \sum_{y \in \{0,1\}^n} f(x \oplus y) g(y),$$

where '$\oplus$' denotes entrywise addition of n-bit strings. If X and Y are independent n-bit random variables, with probability distributions f and g, respectively, then $2^n(f * g)$ is the distribution of the random variable $Z = X \oplus Y$:

$$\Pr[Z = z] = \Pr[X = z \oplus Y] = \sum_{y \in \{0,1\}^n} f(z \oplus y) g(y).$$

This arises naturally in certain computer-science settings, for instance when Y is some error-pattern corrupting X, or when Y is some "mask" used to hide X as in one-time pad encryption.

An important property is that the Fourier transform of the convolution $f * g$ is the product of the Fourier transforms of f and g. This is easily verified by writing out the definitions:

$$\begin{aligned}
\widehat{f * g}(S) &= \frac{1}{2^n}\sum_x (f * g)(x)\chi_S(x) \\
&= \frac{1}{2^{2n}}\sum_x \sum_y f(x \oplus y)g(y)\chi_S(x) \\
&= \frac{1}{2^n}\sum_y g(y)\chi_S(y)\left(\frac{1}{2^n}\sum_x f(x \oplus y)\chi_S(x \oplus y)\right) \\
&= \widehat{f}(S) \cdot \widehat{g}(S).
\end{aligned}$$

Suppose we have two functions $h_0 = f * g_0$ and $h_1 = f * g_1$, for instance from applying the same noise-process f to distributions g_0 and g_1. Using the convolution, we can rewrite their distance as

$$||h_0 - h_1||_2^2 = ||f * (g_0 - g_1)||_2^2 = \sum_S (\widehat{f * (g_0 - g_1)})(S)^2 = \sum_S \widehat{f}(S)^2 \cdot (\widehat{g_0}(S) - \widehat{g_1}(S))^2.$$

This allows us to bound the difference between h_0 and h_1 by analyzing $\widehat{f}$ and $\widehat{g_0} - \widehat{g_1}$.

3 Some elementary applications

3.1 Approximating functions by parities

It is often useful to approximate complex objects by much simpler ones. Consider any function $f : \{0,1\}^n \to [-1,1]$. Suppose there exists a function $p : \{0,1\}^n \to [-1,1]$ that has some positive correlation $\langle f, p\rangle \geq \varepsilon$ with f, and suppose p is *sparse* in that the set $\mathcal{C} = \{S \mid \widehat{p}(S) \neq 0\}$ of nonzero Fourier coefficients has at most c elements. A typical example is where f is a Boolean function and p a low-degree real polynomial approximating f, since degree-d polynomials have at most $c = \sum_{i=0}^{d} \binom{n}{i}$ nonzero Fourier coefficients. Now there exists a parity function that has nontrivial correlation with f, as follows:

$$\varepsilon \leq \langle f, p\rangle = \sum_S \widehat{p}(S)\langle f, \chi_S\rangle \leq ||p||_2 \sqrt{\sum_{S \in \mathcal{C}} \langle f, \chi_S\rangle^2},$$

where we used Cauchy-Schwarz and Parseval. This implies there exists an S such that either χ_S or its negation has correlation $|\langle f, \chi_S\rangle| \geq \varepsilon/(||p||_2\sqrt{c}) \geq \varepsilon/\sqrt{c}$ with f.

3.2 List decoding the Hadamard code

Error-correcting codes are important for storing and sending information in a way that protects against errors. Consider the Hadamard code: for a given $S \in \{0,1\}^n$, the codeword $H(S) \in \{\pm 1\}^{2^n}$ is defined as the concatenation of $\chi_S(x)$ for all x, say ordered in lexicographic order. This code has a terrible rate: n-bit strings are blown up exponentially in size. On the other hand, it has excellent distance: any two codewords are at distance exactly $\frac{1}{2}2^n$. This means that we can always uniquely decode the initial string S from a given word $w \in \{\pm 1\}^{2^n}$ that differs from the codeword $H(S)$ in less than 1/4 of the positions. This is easily phrased in terms of Fourier analysis. View w as a function $w : \{0,1\}^n \to \{\pm 1\}$. Note that $\sum_S \widehat{w}(S)^2 = \mathbb{E}[w^2] = 1$ by Parseval. Then w has normalized distance $d(w, H(S)) = 1/2 - \varepsilon$ from codeword $H(S)$ if, and only if, $\widehat{w}(S) = 2\varepsilon$. If the error-rate is less than 1/4 ($\varepsilon > 1/4$) then the original string is the *unique* S satisfying $\widehat{w}(S) > 1/2$. (There cannot be two distinct S and S' with Fourier coefficients larger than 1/2, since then by the triangle inequality we would have the contradiction $1/2 = d(H(S), H(S')) \leq d(H(S), w) + d(w, H(S')) < 1/2$.)

However, as soon as the error-rate is 1/4 or higher, unique decoding is no longer always possible. For instance the word w that consists of $\frac{3}{4}2^n$ 1s followed by $\frac{1}{4}2^n$ -1s could either have come from $H(0^n) = 1^{2^n}$ or from $H(10^{n-1}) = 1^{2^{n-1}}(-1)^{2^{n-1}}$. Surprisingly, we can still do something useful even if the error-rate is very close to 1/2, say $1/2 - \varepsilon$ for small but positive ε: we can output a small list of potential strings that contains the original string S. This is known as *list decoding.* It does not quite tell us what the original string was, but at least narrows it down to a small set of possibilities. The reason is that not too many codewords $H(S)$ can simultaneously be at a distance $\leq 1/2 - \varepsilon$ from w: such S correspond to Fourier coefficients $\widehat{w}(S) \geq 2\varepsilon$, and not too many Fourier coefficients can be that large since their squares sum to 1. More formally:

$$\#\{S : d(H(S), w) \leq 1/2 - \varepsilon\} \leq \frac{1}{4\varepsilon^2} \sum_{S : d(w, H(S)) \leq 1/2 - \varepsilon} \widehat{w}(S)^2 \leq \frac{1}{4\varepsilon^2} \sum_S \widehat{w}(S)^2 = \frac{1}{4\varepsilon^2}.$$

Note that this upper bound on the list size is independent of n. For instance, if we have error-rate 0.49 (so w is close to random), then still the list of potential codewords has at most $\frac{1}{4(0.01)^2} = 2500$ elements. In fact, Goldreich and Levin [GL89] showed that we can efficiently, in time poly$(n, 1/\varepsilon)$, *find* this list, given oracle access to the 2^n-bit string w. This was the first non-trivial list-decoding algorithm. Later work by Sudan, Guruswami, and others showed similar results for many codes with good rates, see for instance [Sud00, Gur07] and the references therein.

3.3 Learning under the uniform distribution

A lot of work in the last two decades has used the Fourier transform for *learning* under the uniform distribution. The idea is that we can learn an unknown function $f : \{0,1\}^n \to \mathbb{R}$ by approximating its Fourier coefficients. Since the Fourier coefficient

$$\widehat{f}(S) = \mathbb{E}[f \cdot \chi_S]$$

is just an expectation under the uniform distribution on $\{0,1\}^n$, we can approximate it from uniformly drawn examples $(x_1, f(x_1)), \ldots, (x_m, f(x_m))$. The empirical average

$$\frac{1}{m} \sum_{j=1}^{m} f(x_j) \cdot \chi_S(x_j)$$

will converge to the right value $\widehat{f}(S)$ as m grows, and the Chernoff bound tells us how quickly this convergence happens.

Suppose we know f is dominated by a few large Fourier coefficients, and we know those coefficients are contained in a not too large set. A typical case is where f can be approximated by a real polynomial of low degree d. Then we can approximate those coefficients quickly with a small sample size m, and hence learn a good approximation of f. If α_S is our estimate for $\widehat{f}(S)$ and $h = \sum_S \alpha_S \chi_S$ the hypothesis that we output, then by Parseval our overall ℓ_2-error is

$$||f - h||_2^2 = \sum_S (\widehat{f}(S) - \alpha_S)^2.$$

When f has range $\{\pm 1\}$, we can use $\text{sign}(h)$ as our hypothesis for f. We have

$$\Pr[f(x) \neq \text{sign}(h(x))] = \frac{1}{2^n} \sum_{x: f(x) \neq \text{sign}(h(x))} 1 \leq \frac{1}{2^n} \sum_{x: f(x) \neq \text{sign}(h(x))} (f(x) - h(x))^2 \leq ||f - h||_2^2.$$

These ideas have been used for learning constant-depth circuits [LMN93], for learning DNF [Man95, Jac97, KS04, BMOS05], juntas [MOS04], decision trees [KM93, OS07], and others.

4 The Bonami-Beckner inequality and some of its consequences

4.1 Bonami-Beckner and KKL

Consider a function $f : \{0,1\}^n \to \mathbb{R}$. Suppose its input $x \in \{0,1\}^n$ is "noisy": a new input y is obtained by flipping, independently, each bit of x with a fixed probability $\varepsilon \in [0,1]$. The resulting noisy version of f is

$$\widetilde{f}(x) = \mathbb{E}_y[f(y)].$$

Noise has a smoothing effect: sharp peaks in f will be "smeared out" over nearby inputs. Consider for instance a function that is non-zero only on input 0^n: $f(0^n) = 1$ and $f(x) = 0$ for all $x \neq 0^n$. Then $\widetilde{f}$ is a fairly smooth probability distribution "around" 0^n: $\widetilde{f}(x) = \varepsilon^{|x|}(1-\varepsilon)^{n-|x|}$. If $\varepsilon \in (0, 1/2)$ then the maximum of $\widetilde{f}$ still occurs at 0^n, but $\widetilde{f}$ is much less sharply peaked than the original f.

Now consider the linear map that takes f to $\widetilde{f}$. Applying this map to the function f defined by $f(x) = (-1)^{x_i}$ gives $\widetilde{f}(x) = (1-\varepsilon)(-1)^{x_i} + \varepsilon(-1)^{1-x_i} = (1-2\varepsilon)(-1)^{x_i}$. Similarly, applying it to χ_S gives $(1-2\varepsilon)^{|S|}\chi_S$. Hence our map is a function T_ρ that merely shrinks the Fourier coefficient $\widehat{f}(S)$ by a factor $\rho^{|S|}$, where $\rho = 1 - 2\varepsilon \in [-1, 1]$:

$$\widetilde{f} = T_\rho(f) = \sum_S \rho^{|S|} \widehat{f}(S) \chi_S.$$

Here we can see the smoothing effect in action: T_ρ attenuates the higher-degree Fourier coefficients, thus moving f closer to a constant function. For $\rho < 1$, the constant functions are the only ones satisfying $T_\rho(f) = f$.

Generalizing the ℓ_2-norm, for every $p \geq 1$ we define the p-norm of a function by $||f||_p = (\frac{1}{2^n} \sum_x |f(x)|^p)^{1/p}$. One can show this is monotone non-decreasing in p. Since $T_\rho(f)$ is an average of functions that all have the same p-norm as f, the triangle inequality immediately implies that T_ρ is a contraction: for every $p \geq 1$ we have $||T_\rho(f)||_p \leq ||f||_p$. The *Bonami-Beckner Hypercontractive Inequality* [Bon70, Gro75, Bec75] says that this inequality remains true even if we increase the left-hand side by going to a somewhat higher q-norm:

Theorem 1 (Bonami-Beckner) *If* $1 \leq p \leq q$ *and* $0 \leq \rho \leq \sqrt{(p-1)/(q-1)}$, *then*

$$||T_\rho(f)||_q \leq ||f||_p.$$

This inequality is a crucial tool in most of the more advanced applications of Fourier analysis on the Boolean cube. The case $\rho = \sqrt{(p-1)/(q-1)}$ is the strongest case, and implies all others by monotonicity of the p-norm. Its proof is by induction on n. The base case ($n = 1$) is actually the harder part of the induction. We refer to Lecture 16 of [O'D07] for a proof as well as some background and history. Chapter 5 of the book of Janson [Jan97] gives a more general treatment of hypercontractivity.

For us the most interesting cases are when either p or q equal 2, since Parseval allows us to rewrite the 2-norm in terms of Fourier coefficients. This leads to interesting statements about the relations between various norms of f. For instance, suppose the degree of f is at most d. Then for all $p \in [1, 2]$, and using $q = 2$ and $\rho = \sqrt{p-1}$, we have

$$(p-1)^d ||f||_2^2 = (p-1)^d \sum_S \widehat{f}(S)^2 \leq \sum_S (p-1)^{|S|} \widehat{f}(S)^2 = ||T_{\sqrt{p-1}}(f)||_2^2 \leq ||f||_p^2.$$

Hence the p-norm of a low-degree function cannot be much smaller than its 2-norm. A similar argument gives $||f||_q^2 \leq (q-1)^d ||f||_2^2$ for all $q \geq 2$.

In general, with $q = 2$, $p \in [1, 2]$, and $\rho = \sqrt{p-1}$, Theorem 1 becomes

$$\sum_S (p-1)^{|S|} \widehat{f}(S)^2 = ||T_\rho(f)||_2^2 \leq ||f||_p^2 = \left(\frac{1}{2^n} \sum_x |f(x)|^p \right)^{2/p}. \qquad (2)$$

This gives an upper bound on the squared Fourier coefficients of f, in a way that gives most weight to the low-degree coefficients: each coefficient is "weighed down" by $(p-1)^{|S|}$.[1] An important special case of Eq. (2) is where f has range $\{-1, 0, 1\}$. This occurs for instance if f is a Boolean function or the difference of two Boolean functions. In that case we have $||f||_p^p = \Pr[f \neq 0]$ for any p. Applying Eq. (2) with $p = 1 + \delta$ gives the following *KKL Inequality*, after Kahn, Kalai, and Linial [KKL88].

Corollary 1 (KKL) *For every $\delta \in [0, 1]$ and $f : \{0,1\}^n \to \{-1, 0, 1\}$, we have*

$$\sum_S \delta^{|S|} \widehat{f}(S)^2 \leq (\Pr[f \neq 0])^{2/(1+\delta)}.$$

Informally, with $\delta < 1$ the left-hand side is dominated by the Fourier coefficients of low degree (i.e., those where $|S|$ is small). The right-hand side is smaller than the total "Fourier weight" $\sum_S \widehat{f}(S)^2 = \Pr[f \neq 0]$ by a power $2/(1+\delta) > 1$. Hence the inequality says that a $\{-1, 0, 1\}$-valued function with small support cannot have too much of its Fourier weight on low degrees.

4.2 Random parities over a fixed set

An application for which KKL seems to be almost tailor-made, is to bound the expected bias of k-bit parities over a set $A \subseteq \{0,1\}^n$. Suppose we pick a set $S \subseteq [n]$ of k indices uniformly at random, and consider the parity of the k-bit substring induced by S and a uniformly random $x \in A$. Intuitively, if A is large then we expect that for most S, the bias β_S of this parity to be small: the number of $x \in A$ with $\chi_S(x) = 1$ should be roughly the same as with $\chi_S(x) = -1$. This setting is relevant to cryptography. Suppose we have an n-bit string x about which our adversary

[1] Recently Eq. (2) was generalized from real-valued functions to *matrix-valued* functions [BRW08].

has some limited knowledge: he only knows that x is uniformly distributed over some fairly large set A. Then our adversary will be unable to predict most parities of selected bits from x, and we can use such parities to obtain bits that are essentially unknown to him. We ourselves do not need to know A for this; we only need to know a lower bound on its size, i.e., an upper bound on the adversary's knowledge.

The intuition that large A leads to small biases is justified by the KKL Inequality.[2] Note the connection between biases and Fourier coefficients: with f the characteristic function of A, we have

$$\beta_S = \mathbb{E}_{x\in A}[\chi_S(x)] = \frac{1}{|A|}\sum_{x\in A}\chi_S(x) = \frac{1}{|A|}\sum_{x\in\{0,1\}^n} f(x)\chi_S(x) = \frac{2^n}{|A|}\widehat{f}(S).$$

Applying the KKL Inequality, for any $\delta \in [0,1]$ we can bound the sum of *squared* biases by

$$\sum_{S\in\binom{[n]}{k}} \beta_S^2 = \frac{2^{2n}}{|A|^2}\sum_{S\in\binom{[n]}{k}} \widehat{f}(S)^2 \le \frac{2^{2n}}{\delta^k|A|^2}\left(\frac{|A|}{2^n}\right)^{2/(1+\delta)} \le \frac{1}{\delta^k}\left(\frac{2^n}{|A|}\right)^{2\delta}.$$

By differentiating, one can show that $\delta = \frac{k}{2\ln(2^n/|A|)}$ minimizes the right-hand side (assume $k \le 2\ln(2^n/|A|)$ to ensure $\delta \in [0,1]$). This gives

$$\mathbb{E}_S[\beta_S^2] = \frac{1}{\binom{n}{k}}\sum_{S\in\binom{[n]}{k}} \beta_S^2 = O\left(\frac{\log(2^n/|A|)}{n}\right)^k. \tag{3}$$

The following example shows this bound is essentially tight, and hence the KKL Inequality is tight as well. Let $A = 0^c \times \{0,1\}^{n-c}$ consist of all 2^{n-c} strings starting with c zeroes. Then $\beta_S = 1$ if $S \subseteq [c]$, and $\beta_S = 0$ otherwise. The fraction of sets $S \in \binom{[n]}{k}$ satisfying $S \subseteq [c]$ is $\binom{c}{k}/\binom{n}{k} = \Omega(c/n)^k$. Hence $\mathbb{E}_S[\beta_S^2] = \Omega(c/n)^k = \Omega(\log(2^n/|A|)/n)^k$, matching the upper bound of Eq. (3).

4.3 Influences of variables

Suppose we have n players, and we have a function $f : \{0,1\}^n \to \{0,1\}$ where player i controls the bit x_i. If f is balanced (meaning exactly half of the 2^n inputs x yield $f(x) = 1$), then we can use it to implement a *collective coin flipping scheme*: let each player pick their bit x_i randomly and use $f(x)$ as the value of the coin flip. If all players indeed follow this protocol, the result is a fair coin flip. However, in order for this to be secure, small collusions of cheating players who can see the bits of the honest players should not be able to influence the function's output value too much.

Formally, the *influence* of variable $i \in [n]$ on f is defined as

$$\mathrm{Inf}_i(f) = \Pr[f(x) \ne f(x \oplus e_i)],$$

where the probability is over uniform $x \in \{0,1\}^n$, and $x \oplus e_i$ is x with the ith bit flipped. This measures the probability (over random bits for all other players) that player i can determine the function value. One can generalize this definition to the influence of a *set* S of players in the

[2] To the best of our knowledge, the bound below was first shown by Talagrand [Tal96, Eq. (2.9)] for $k = 2$, using a large deviation inequality instead of hypercontractivity, and for general k in [GKK+07] using the KKL Inequality. [GKK+07] used it to prove lower bounds for communication complexity.

obvious way: $\mathrm{Inf}_S(f)$ is the probability that the function is non-constant when all variables outside of S are set randomly. Two extreme cases are the constant function (where $\mathrm{Inf}_i(f) = 0$ for all i), and the parity function (where $\mathrm{Inf}_i(f) = 1$ for all i). For the dictator function $f(x) = x_i$, the ith variable has influence 1 while all other influences are 0. Another important example is the n-bit majority function, which is 1 if more than half of its input bits are 1, and which is 0 otherwise. Here each variable has influence $\Theta(1/\sqrt{n})$, because the probability that the other $n-1$ bits are set such that x_i determines the majority value is exactly $\binom{n-1}{\lfloor n/2\rfloor}/2^{n-1}$. Moreover, any set S of, say, $10\sqrt{n}$ variables will with high probability be able to "tip the balance" and determine the majority value when the other $n - 10\sqrt{n}$ input bits are set randomly, hence $\mathrm{Inf}_S(f) \approx 1$.

Can we find balanced functions where the influences of the variables are much smaller, ideally $O(1/n)$? Ben-Or and Linial [BL89] showed that the "tribes" function (an OR-AND tree with bottom fan-out about $\log n - \log\log n$) is a balanced function where every variable has influence $\Theta(\log(n)/n)$. Kahn, Kalai, and Linial [KKL88] later showed that this is essentially optimal: for every Boolean function f that is balanced or close to balanced, at least one of its variables has influence $\Omega(\log(n)/n)$. With some extra work (which we won't detail here), this implies the existence of a set of only $O(n/\log n)$ variables that together determine the function value for almost all settings of the other variables. Hence we cannot hope to use such functions for collective coin flipping protocols that are secure against a constant fraction of cheating players: a small set of $O(n/\log n)$ colluding players can already control the outcome with high probability.

The KKL result was one of the first major applications of Fourier analysis to Boolean functions. We will prove a slightly stronger result here due to Talagrand [Tal94]. Let $\mathrm{Var}[f] = \mathbb{E}[f^2] - \mathbb{E}[f]^2$ be the *variance* of f. By Parseval, this equals $||f||_2^2 - \widehat{f}(\emptyset)^2 = \sum_{S\neq\emptyset} \widehat{f}(S)^2$. Consider a function $f : \{0,1\}^n \to \{0,1\}$. Assume that no variable has influence exactly 0 or 1 (influence-0 variables are irrelevant anyway, and influence-1 variables can be "factored" out of the function). Then we have

$$\sum_{i=1}^{n} \frac{\mathrm{Inf}_i(f)}{\log(1/\mathrm{Inf}_i(f))} = \Omega(\mathrm{Var}[f]). \tag{4}$$

The KKL result follows immediately from this: if f is close to balanced then $\mathrm{Var}[f] = \Omega(1)$. Hence there is an i such that $\mathrm{Inf}_i(f)/\log(1/\mathrm{Inf}_i(f)) = \Omega(1/n)$, which implies $\mathrm{Inf}_i(f) = \Omega(\log(n)/n)$.

The proof of Eq. (4) is based on the following technical lemma, which uses Bonami-Beckner.

Lemma 1 *If $g : \{0,1\}^n \to \mathbb{R}$ satisfies $||g||_{3/2} \neq ||g||_2$, then* $\displaystyle\sum_{S\neq\emptyset} \frac{\widehat{g}(S)^2}{|S|} \leq \frac{2.5\,||g||_2^2}{\log(||g||_2/||g||_{3/2})}$.

Proof. Using Theorem 1 with $q = 2$ and $p = 1 + \rho^2 = 3/2$ we get for every integer k

$$\sum_{S:|S|=k} \widehat{g}(S)^2 \leq 2^k \sum_S 2^{-|S|}\widehat{g}(S)^2 = 2^k ||T_{\sqrt{1/2}}(g)||_2^2 \leq 2^k ||g||_{3/2}^2.$$

For every integer m we have (using $\sum_{k=1}^m 2^k/k \leq 4\cdot 2^m/(m+1)$, which is easily proved by induction):

$$\sum_{S\neq\emptyset} \frac{\widehat{g}(S)^2}{|S|} = \sum_{k=1}^{m} \sum_{S:|S|=k} \frac{\widehat{g}(S)^2}{k} + \sum_{S:|S|>m} \frac{\widehat{g}(S)^2}{|S|} \leq \sum_{k=1}^{m} \frac{2^k ||g||_{3/2}^2}{k} + \sum_{S:|S|>m} \frac{\widehat{g}(S)^2}{m+1} \leq \frac{4\cdot 2^m ||g||_{3/2}^2 + ||g||_2^2}{m+1}$$

Choose m the largest integer satisfying $2^m||g||_{3/2}^2 \leq ||g||_2^2$. Then $m+1 > 2\log(||g||_2/||g||_{3/2})$, and

$$\sum_{S\neq\emptyset} \frac{\widehat{g}(S)^2}{|S|} \leq \frac{5\,||g||_2^2}{m+1} \leq \frac{2.5\,||g||_2^2}{\log(||g||_2/||g||_{3/2})}.$$

□

Now consider a variable i and define $g(x) = f(x) - f(x \oplus e_i)$. Then $g(x) = 0$ if $f(x) = f(x \oplus e_i)$, and $g(x) \in \{\pm 1\}$ otherwise. Hence $||g||_2^2 = ||g||_{3/2}^{3/2} = \mathrm{Inf}_i(f) \in (0,1)$, and $||g||_2/||g||_{3/2} = \mathrm{Inf}_i(f)^{-1/6}$. The Fourier coefficients of g are closely related to those of f:

$$\widehat{g}(S) = \begin{cases} 2\widehat{f}(S) & \text{if } i \in S, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Applying Lemma 1 to g gives

$$\sum_{S:i\in S} \frac{4\widehat{f}(S)^2}{|S|} = \sum_{S} \frac{\widehat{g}(S)^2}{|S|} \leq \frac{2.5\,||g||_2^2}{\log(||g||_2/||g||_{3/2})} = \frac{15\,\mathrm{Inf}_i(f)}{\log(1/\mathrm{Inf}_i(f))}.$$

Summing over all i gives Talagrand's result:

$$4\mathrm{Var}[f] = 4\sum_{S\neq 0} \widehat{f}(S)^2 = \sum_{i=1}^{n} \sum_{S:i\in S} \frac{4\widehat{f}(S)^2}{|S|} \leq \sum_{i=1}^{n} \frac{15\,\mathrm{Inf}_i(f)}{\log(1/\mathrm{Inf}_i(f))}.$$

Subsequent work. The subsequent "BKKKL" paper [BKK$^+$92] generalized the KKL result to the case of functions $f : [0,1]^n \to \{0,1\}$ with real-valued input, with uniform measure on each real-valued variable x_i. See Friedgut [Fri04] for some simplifications and corrections of this. A recent related result is that any near-balanced Boolean function with a depth-d decision tree has a variable with influence $\Omega(1/d)$ [OSSS05].

4.4 The relation between influences, sensitivity, and degree

In this section we relate influences to degree and sensitivity, both of which are important measures of the complexity of Boolean functions.[3] This section does not need the Bonami-Beckner or KKL Inequalities, but is placed after Section 4.3 because it considers the influences defined there. The *sensitivity* of a function $f : \{0,1\}^n \to \{0,1\}$ on input x is

$$s_x(f) = |\{i \mid f(x) \neq f(x \oplus e_i)\}|$$

and the *average sensitivity* of f is

$$\overline{s}(f) = \frac{1}{2^n} \sum_{x\in\{0,1\}^n} s_x(f).$$

[3] See the survey [BW02]. Other Fourier-based results on polynomial degrees are in [OS03, Ser07, BVW07].

By linearity of expectation, average sensitivity equals the total influence:

$$\overline{s}(f) = \sum_{i=1}^{n} \mathrm{Inf}_i(f). \tag{6}$$

Here we give a relation between degree and sensitivity due to Shi [Shi00]: average sensitivity lower bounds the degree of approximating polynomials. Suppose a degree-d n-variate real polynomial $p : \{0,1\}^n \to [0,1]$ approximates $f : \{0,1\}^n \to \{0,1\}$, in the sense that there is an $\varepsilon \in [0,1/2)$ such that $|f(x) - p(x)| \leq \varepsilon$ for every $x \in \{0,1\}^n$. Let q be the degree-d polynomial $1 - 2p$. This has range $[-1,1]$, hence $\sum_S \widehat{q}(S)^2 = ||q||_2^2 \leq 1$. Note that $q(x) \in [-1, -1+2\varepsilon]$ if $f(x) = 1$, and $q(x) \in [1-2\varepsilon, 1]$ if $f(x) = 0$. Consider the function $q^{(i)}(x) = q(x) - q(x \oplus e_i)$. Using Parseval and the analogue of Eq. (5), we have

$$(2-4\varepsilon)^2 \mathrm{Inf}_i(f) \leq \mathbb{E}[(q^{(i)})^2] = \sum_S \widehat{q^{(i)}}(S)^2 = 4 \sum_{S: i \in S} \widehat{q}(S)^2.$$

Dividing by 4 and summing over all i gives the lower bound on the degree:

$$(1-2\varepsilon)^2 \overline{s}(f) = (1-2\varepsilon)^2 \sum_{i=1}^{n} \mathrm{Inf}_i(f) \leq \sum_{i=1}^{n} \sum_{S: i \in S} \widehat{q}(S)^2 = \sum_S |S| \widehat{q}(S)^2 \leq d \sum_S \widehat{q}(S)^2 \leq d.$$

A random Boolean function has $\overline{s}(f) \approx n/2$ and ε-approximate degree at least $n/2 - O(\sqrt{n})$ [Amb99] (for fixed ε), hence this bound is optimal up to constant factors for almost all n-bit functions.

Another Fourier-based lower bound on the degree is due to Nisan and Szegedy [NS94]. Suppose $f : \{0,1\}^n \to \{0,1\}$ depends on all of its n variables (equivalently, each variable has positive influence). Consider $f^{(i)}(x) = f(x) - f(x \oplus e_i)$. This is an n-variate polynomial of degree $d \leq \deg(f)$, and it is non-constant because f depends on x_i. It is well known that such a polynomial is non-zero on at least a 2^{-d}-fraction of its inputs.[4] Hence

$$\mathrm{Inf}_i(f) = \Pr[f^{(i)} \neq 0] \geq 2^{-\deg(f)}$$

for each i. Summing over i we get

$$\frac{n}{2^{\deg(f)}} \leq \sum_{i=1}^{n} \mathrm{Inf}_i(f) = \sum_{i=1}^{n} \sum_S \widehat{f^{(i)}}(S)^2 = 4 \sum_S |S| \widehat{f}(S)^2 \leq 4 \deg(f) \sum_S \widehat{f}(S)^2 \leq 4 \deg(f).$$

This implies the bound from [NS94]:

$$\deg(f) \geq \log(n) - O(\log\log n)$$

for every Boolean function f that depends on n variables. As Nisan and Szegedy observed, the "address function" shows that this bound is tight up to the $O(\log\log n)$ term. This function takes an input $x_1 \ldots x_m y_0 \ldots y_{2^m-1}$ of $n = m + 2^m$ bits, and outputs y_i where i is the number whose binary representation is $x_1 \ldots x_m$. The function depends on all n variables. It is represented by $\sum_{i \in \{0,1\}^m} y_i \prod_{j: i_j = 1} x_j \prod_{j: i_j = 0} (1 - x_j)$ and hence has degree $m + 1 \leq \log n + 1$.

[4] This is a basic and well known property of Reed-Muller error-correcting codes [MS77]. In the computer science literature this fact is usually called the Schwartz-Zippel Lemma [Sch80, Zip07].

5 A guide to literature

The examples given above illustrate the usefulness of Fourier analysis on the Boolean cube, but they barely scratch the surface of the rich set of actual and potential applications. In this last section we give pointers to the other main areas of application in computer science.

PCPs and hardness of approximation. Possibly the most important application of Fourier analysis in theoretical computer science is its use in designing and analyzing *Probabilistically Checkable Proofs*. These are encodings of witnesses for NP-like problems that can be verified probabilistically while querying only a small number of their bits. The famous PCP Theorem [ALM$^+$98, Din07] says that a language is in NP if, and only if, it has witnesses that can be verified probabilistically using only $O(\log n)$ bits of randomness and a *constant* number of queries to bits of the witness. Some of the most efficient PCPs are based on Fourier analysis, Håstad's 3-query PCP being a prime example [Hås01]. Based on these PCPs, one can show NP-hardness results for approximations to many optimization problems, such as finding a maximal clique in a graph or the maximal number of satisfied clauses in a given CNF formula. Chapters 11 and 22 of the book by Arora and Barak [AB09] give a good introduction to this material.

A recent development is the use of the "Unique Games Conjecture." This conjecture, due to Khot [Kho02], says that it is hard to approximate the maximal number of satisfied constraints in problems with only two variables per constraint, where the value of either variable in a constraint determines the value of the other (the variables are over a sufficiently large but constant domain). Assuming this, one can prove essentially optimal inapproximability results for problems like max-cut [KKMO07], vertex cover [KR08], and others [CKK$^+$05, KV05, DMR06, KO06, Rag08], which so far resisted the more standard inapproximability approach via PCPs. Again, Fourier techniques are often essential in the analysis. For instance, one of the main planks of the max-cut result is the "Majority Is Stablest" Theorem. This was conjectured in a first version of [KKMO07] and proved in [MOO08]. It says that among all balanced Boolean functions where every variable has low influence, the majority function has the maximal correlation between f and its noisy version $\tilde{f} = T_\rho(f)$ (as defined in Section 4.1).

Threshold phenomena. A *threshold phenomenon* occurs if certain properties of a system change sharply in response to a small change of an underlying parameter which is close to a specific value (the "threshold"). A typical example in nature is water, which is frozen if the temperature-parameter is just below 0°C and liquid if the temperature is just above 0°C. In mathematics, such phenomena occur for instance in random graphs. Let $G(n, p)$ denote an undirected graph on n vertices, where each edge is included with probability p (independent of the other edges). Erdős and Rényi [ER59] introduced this model and showed a sharp threshold for connectivity: if p is slightly below $(\log n)/n$ then the probability of $G(n, p)$ being connected tends to 0 with n, while if p is slightly larger than $(\log n)/n$ then this probability tends to 1. Friedgut and Kalai [FK96] later showed that *every* monotone graph property has a sharp threshold.

Threshold phenomena occur also in complexity theory. For instance, if one picks a random 3-SAT formula with n variables and $m = cn$ clauses, then for $c < 4.25$ (roughly), the formula is most probably satisfiable, while for $c > 4.25$ the formula is most probably unsatisfiable.[5] Similarly, one

[5] Actually this value of 4.25 is a numerical estimate; proven upper and lower bounds on this number are far from tight. We refer to Part 3 of [PIM06] for details.

can view tight results on hardness of approximation as threshold phenomena: for many NP-hard optimization problems, there exists a constant c such that approximating the optimal value up to factor c can be done in polynomial time, while approximating it to within a slightly better factor is NP-hard (or Unique-Games hard). Kalai and Safra [KS06] give a very interesting survey of these phenomena, showing how influences and Fourier techniques are central to their analysis. Often these techniques apply to the generalized setting where each input bit is 1 with probability p.

Social choice theory. If n people have to decide between two alternatives then they can use majority voting, which has all the properties one expects of a reasonable voting scheme. However, as soon as they have to choose between three or more alternatives, Arrow's Theorem [Arr50] says that no "ideal" voting scheme exists. Strong quantitative versions of this theorem can be obtained quite easily using Fourier techniques. The surveys by Kalai [Kal05] and O'Donnell [O'D08] are excellent starting points for such connections between Fourier analysis and social choice theory. A very recent related result is [FKN08].

When are functions close to juntas? Recall that a k-junta is a function on $\{0,1\}^n$ that depends on at most k of its variables. Friedgut [Fri98] showed that if the average sensitivity (a.k.a. total influence) of a Boolean function f is I, then f is close to another Boolean function that is a $2^{O(I)}$-junta. The "address function" from Section 4.4 shows the exponential is necessary: it has average sensitivity $O(\log n)$ but cannot be approximated well by a Boolean function depending on only $o(n)$ variables. Friedgut et al. [FKN02] show that if the Fourier coefficients of degrees 0 and 1 have most of the total weight, then f is close to a 1-junta (i.e., a dictator or its negation). Bourgain [Bou02] proved that the weight on higher-degree Fourier coefficients of a balanced Boolean function cannot decay too fast unless f is close to a junta, and Dinur et al. [DFKO07] analyzed the same phenomenon for *bounded*—but not necessarily Boolean—functions on the Boolean cube.

Other applications. Some results in the area of *property testing* are based on Fourier analysis. Examples are the algorithm by Fischer et al. [FKR$^+$04] for testing if a Boolean function is close to or far from a k-junta, and the one by Alon et al. [AAK$^+$07] for testing if a distribution is close to or far from (almost) k-wise independence. The above-mentioned work on PCPs also falls in this category, since one is basically testing whether a given witness is close to a valid proof or not. In addition to the list-decoding example from Section 3.2, there have been a number of other applications of Fourier analysis in coding theory, see for instance Section 4.3 of Linial's course notes (mentioned below) and Navon and Samorodnitsky [NS05]. Fourier analysis has also been used for lower bounds on various kinds of communication complexity [Raz95, Kla01, GKK$^+$07, She08], and for analysis of low-distortion embeddings of one metric space into another [LN04, KV05].

Other expository papers and courses. Several more extensive surveys on Fourier analysis of Boolean functions exist in the literature. The early one by Bernasconi, Codenotti, and Simon [BCS97] describes the main applications up to 1997. Štefankovič's MSc thesis [Šte00] is geared towards general applications of Fourier analysis in computer science, often over groups other than the Boolean cube. The survey by Kalai and Safra [KS06] is geared towards threshold phenomena. The very recent survey by O'Donnell [O'D08] focuses on topics related to voting and hardness of approximation, and also tries to demystify the Bonami-Beckner Inequality by presenting it as a generalization of the Hoeffding-Chernoff bounds to higher-degree functions. Finally, let us point to the notes of a number of recent courses, which contain a wealth of additional material:

Irit Dinur and Ehud Friedgut: `http://www.cs.huji.ac.il/~analyt`
Subhash Khot: `http://www.cc.gatech.edu/~khot/Fourier.htm`
Guy Kindler: `http://dimacs.rutgers.edu/~gkindler/boolean-course`
Nati Linial: `http://www.cs.huji.ac.il/~nati/PAPERS/uw`
Elchanan Mossel: `http://www.stat.berkeley.edu/%7Emossel/teach/206af05`
Ryan O'Donnell: `http://www.cs.cmu.edu/~odonnell/boolean-analysis`
Oded Regev: `http://www.cs.tau.ac.il/~odedr/teaching/analysis_fall_2007`

Acknowledgments

Thanks to David García Soriano, Elena Grigorescu, Peter Harremoës, Oded Regev, Nitin Saxena, Ben Toner, and Stephanie Wehner for useful comments and pointers. Also thanks to the anonymous ToC referees for many extremely helpful recommendations.

References

[AAK$^+$07] N. Alon, A. Andoni, T. Kaufman, K. Matulef, R. Rubinfeld, and N. Xie. Testing k-wise and almost k-wise independence. In *Proceedings of 39th ACM STOC*, pages 496–505, 2007.

[AB09] S. Arora and B. Barak. *Complexity Theory: A Modern Approach*. Cambridge University Press, 2009. To appear. Preliminary version available at `http://www.cs.princeton.edu/theory/complexity`

[ALM$^+$98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998. Earlier version in FOCS'92.

[Amb99] A. Ambainis. A note on quantum black-box complexity of almost all Boolean functions. *Information Processing Letters*, 71(1):5–7, 1999.

[Arr50] K. Arrow. A difficulty in the concept of social welfare. *Journal of Political Economy*, 58(4):328–346, 1950.

[BCS97] A. Bernasconi, B. Codenotti, and J. Simon. On the Fourier analysis of Boolean functions. Technical Report IMC B4-97-03, Istituto di Matematica Computazionale, Pisa, 1997.

[Bec75] W. Beckner. Inequalities in Fourier analysis. *Annals of Mathematics*, 102:159–182, 1975.

[BKK$^+$92] J. Bourgain, J. Kahn, G. Kalai, G. Katznelson, and N. Linial. The influence of variables in product spaces. *Israel Journal of Mathematics*, 77:55–64, 1992.

[BL89] M. Ben-Or and N. Linial. Collective coin flipping. In S. Micali, editor, *Randomness and Computation*, pages 91–115. JAI Press, 1989. Earlier version in FOCS'85.

[BMOS05] N. Bshouty, E. Mossel, R. O'Donnell, and R. Servedio. Learning DNF from random walks. *Journal of Computer and System Sciences*, 71(3):250–265, 2005. Earlier version in FOCS'03.

[Bon70] A. Bonami. Étude des coefficients de Fourier des fonctions de $L^p(G)$. *Annales de l'Institut Fourier*, 20(2):335–402, 1970.

[Bou02] J. Bourgain. On the distribution of the Fourier spectrum of Boolean functions. *Israel Journal of Mathematics*, 131(1):269–276, 2002.

[BRW08] A. Ben-Aroya, O. Regev, and R. de Wolf. A hypercontractive inequality for matrix-valued functions with applications to quantum computing. In *Proceedings of 49th IEEE FOCS*, 2008.

[BVW07] H. Buhrman, N. Vereshchagin, and R. de Wolf. On computation and communication with small bias. In *Proceedings of 22nd IEEE Conference on Computational Complexity*, pages 24–32, 2007.

[BW02] H. Buhrman and R. de Wolf. Complexity measures and decision tree complexity: A survey. *Theoretical Computer Science*, 288(1):21–43, 2002.

[CKK$^+$05] S. Chawla, R. Krauthgamer, R. Kumar, Y. Rabani, and D. Sivakumar. On the hardness of approximating sparsest cut and multicut. In *Proceedings of 20th IEEE Conference on Computational Complexity*, pages 144–153, 2005.

[DFKO07] I. Dinur, E. Friedgut, G. Kindler, and R. O'Donnell. On the Fourier tails of bounded functions over the discrete cube. *Israel Journal of Mathematics*, 160(1):389–412, 2007. Earlier version in STOC'06.

[Din07] I. Dinur. The PCP theorem by gap amplification. *Journal of the ACM*, 54(3(12)), 2007. Earlier version in STOC'06.

[DMR06] I. Dinur, E. Mossel, and O. Regev. Conditional hardness for approximate coloring. In *Proceedings of 38nd ACM STOC*, pages 344–353, 2006.

[ER59] P. Erdős and A. Rényi. On random graphs I. *Publicationes Mathematicae*, 6:290–297, 1959.

[FK96] E. Friedgut and G. Kalai. Every monotone graph property has a sharp threshold. *Proceedings of the AMS*, 124:2993–3002, 1996.

[FKN02] E. Friedgut, G. Kalai, and A. Naor. Boolean functions whose Fourier transform is concentrated at the first two levels. *Advances in Applied Mathematics*, 29(3):427–437, 2002.

[FKN08] E. Friedgut, G. Kalai, and N. Nisan. Elections can be manipulated often. In *Proceedings of 49th IEEE FOCS*, 2008.

[FKR$^+$04] E. Fischer, G. Kindler, D. Ron, S. Safra, and A. Samorodnitsky. Testing juntas. *Journal of Computer and System Sciences*, 68(4):753–787, 2004. Earlier version in FOCS'02.

[Fri98] E. Friedgut. Boolean functions with low average sensitivity depend on few coordinates. *Combinatorica*, 18(1):27–35, 1998.

[Fri04] E. Friedgut. Influences in product spaces: KKL and BKKKL revisited. *Combinatorics, Probability and Computing*, 13:17–29, 2004.

[GKK$^+$07] D. Gavinsky, J. Kempe, I. Kerenidis, R. Raz, and R. de Wolf. Exponential separations for one-way quantum communication complexity, with applications to cryptography. In *Proceedings of 39th ACM STOC*, pages 516–525, 2007.

[GL89] O. Goldreich and L. Levin. A hard-core predicate for all one-way functions. In *Proceedings of 21st ACM STOC*, pages 25–32, 1989.

[Gro75] L. Gross. Logarithmic Sobolev inequalities. *American Journal of Mathematics*, 97(4):1061–1083, 1975.

[Gur07] V. Guruswami. *Algorithmic Results in List Decoding*, volume 2(2) of *Foundations and Trends in Theoretical Computer Science*. Now Publishers, 2007.

[Hås01] J. Håstad. Some optimal inapproximability results. *Journal of the ACM*, 48(4):798–859, 2001. Earlier version in STOC'97.

[Jac97] J. Jackson. An efficient membership-query algorithm for learning DNF with respect to the uniform distribution. *Journal of Computer and System Sciences*, 55(3):414–440, 1997. Earlier version in FOCS'94.

[Jan97] S. Janson. *Gaussian Hilbert Spaces*, volume 129 of *Cambridge Tracts in Mathematics*. Cambridge University Press, 1997.

[Kal05] G. Kalai. Noise sensitivity and chaos in social choice theory. Technical report, Discussion Paper Series dp399. Center for rationality and interactive decision theory, Hebrew University, 2005. Available at `http://www.ma.huji.ac.il/~kalai/CHAOS.pdf`

[Kho02] S. Khot. On the power of unique 2-prover 1-round games. In *Proceedings of 34th ACM STOC*, pages 767–775, 2002.

[KKL88] J. Kahn, G. Kalai, and N. Linial. The influence of variables on Boolean functions. In *Proceedings of 29th IEEE FOCS*, pages 68–80, 1988.

[KKMO07] S. Khot, G. Kindler, E. Mossel, and R. O'Donnell. Optimal inapproximability results for MAX-CUT and other 2-variable CSPs? *SIAM Journal on Computing*, 37(1):319–357, 2007.

[Kla01] H. Klauck. Lower bounds for quantum communication complexity. In *Proceedings of 42nd IEEE FOCS*, pages 288–297, 2001.

[KM93] E. Kushilevitz and Y. Mansour. Learning decision trees using the Fourier spectrum. *SIAM Journal on Computing*, 22(6):1331–1348, 1993. Earlier version in STOC'91.

[KO06] S. Khot and R. O'Donnell. SDP gaps and UGC-hardness for MAXCUTGAIN. In *Proceedings of 47th IEEE FOCS*, pages 217–226, 2006.

[KR08] S. Khot and O. Regev. Vertex cover might be hard to approximate to within $2 - \varepsilon$. *Journal of Computer and System Sciences*, 74(3):335–349, 2008. Earlier version in CCC'03.

[KS04] A. Klivans and R. Servedio. Learning DNF in time $2^{\tilde{O}(n^{1/3})}$. *Journal of Computer and System Sciences*, 68(2):303–318, 2004. Earlier version in STOC'01.

[KS06] G. Kalai and S. Safra. Threshold phenomena and influence. In Percus et al. [PIM06], pages 25–60.

[KV05] S. Khot and N. Vishnoi. The unique games conjecture, integrality gap for cut problems and embeddability of negative type metrics into ℓ_1. In *Proceedings of 46th IEEE FOCS*, pages 53–62, 2005.

[LMN93] N. Linial, Y. Mansour, and N. Nisan. Constant depth circuits, Fourier transform, and learnability. *Journal of the ACM*, 40(3):607–620, 1993. Earlier version in FOCS'89.

[LN04] J. R. Lee and A. Naor. Embedding the diamond graph in L_p and dimension reduction in L_1. *Geometric and Functional Analysis*, 14(4):745–747, 2004.

[Man95] Y. Mansour. An $O(n^{\log \log n})$ learning algorithm for DNF under the uniform distribution. *Journal of Computer and System Sciences*, 50(3):543–550, 1995. Earlier version in COLT'92.

[MOO08] E. Mossel, R. O'Donnell, and K. Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. *Annals of Mathematics*, 2008. To appear. Earlier version in FOCS'05.

[MOS04] E. Mossel, R. O'Donnell, and R. Servedio. Learning functions of k relevant variables. *Journal of Computer and System Sciences*, 69(3):421–434, 2004. Earlier version in STOC'03.

[MS77] F. MacWilliams and N. Sloane. *The Theory of Error-Correcting Codes*. North-Holland, 1977.

[NS94] N. Nisan and M. Szegedy. On the degree of Boolean functions as real polynomials. *Computational Complexity*, 4(4):301–313, 1994. Earlier version in STOC'92.

[NS05] M. Navon and A. Samorodnitsky. On Delsarte's linear programming bounds for binary codes. In *Proceedings of 46th IEEE FOCS*, pages 327–338, 2005.

[O'D07] R. O'Donnell. Lecture notes for a course "Analysis of Boolean functions", 2007. Available at `http://www.cs.cmu.edu/~odonnell/boolean-analysis`

[O'D08] R. O'Donnell. Some topics in analysis of boolean functions. Technical report, ECCC Report TR08–055, 2008. Paper accompanying an invited talk at STOC'08.

[OS03] R. O'Donnell and R. Servedio. Extremal properties of polynomial threshold functions. In *Proceedings of 18th IEEE Conference on Computational Complexity*, pages 3–12, 2003.

[OS07] R. O'Donnell and R. Servedio. Learning monotone decision trees in polynomial time. *SIAM Journal on Computing*, 37(3):213–225, 2007. Earlier version in Complexity'06.

[OSSS05] R. O'Donnell, M. Saks, O. Schramm, and R. Servedio. Every decision tree has an influential variable. In *Proceedings of 46th IEEE FOCS*, pages 31–39, 2005.

[PIM06] A.G. Percus, G. Istrate, and C. Moore, editors. *Computational Complexity and Statistical Physics*. Oxford University Press, 2006.

[Rag08] P. Raghavendra. Optimal algorithms and inapproximability results for every CSP? In *Proceedings of 40th ACM STOC*, pages 245–254, 2008.

[Raz95] R. Raz. Fourier analysis for probabilistic communication complexity. *Computational Complexity*, 5(3/4):205–221, 1995.

[Sch80] J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *Journal of the ACM*, 27:701–717, 1980.

[Ser07] R. Servedio. Every linear threshold function has a low-weight approximator. *Computational Complexity*, 16(2):180–209, 2007. Earlier version in Complexity'06.

[She08] A. Sherstov. Communication lower bounds using dual polynomials. Technical report, ECCC Report TR08–057, 2008.

[Shi00] Y. Shi. Lower bounds of quantum black-box complexity and degree of approximating polynomials by influence of Boolean variables. *Information Processing Letters*, 75(1–2):79–83, 2000.

[Šte00] D. Štefankovič. Fourier transforms in computer science. Master's thesis, University of Chicago, Department of Computer Science, 2000. Available at `http://www.cs.rochester.edu/~stefanko/Publications/Fourier.ps`

[Sud00] M. Sudan. List decoding: Algorithms and applications. In *Proceedings of the International Conference IFIP TCS*, volume 1872 of *Lecture Notes in Computer Science*, pages 25–41. Springer, 2000.

[Tal94] M. Talagrand. On Russo's approximate 0-1 law. *Annals of Probability*, 22:1576–1587, 1994.

[Tal96] M. Talagrand. How much are increasing sets positively correlated? *Combinatorica*, 16(2):243–258, 1996.

[Zip07] R. E. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of EUROSAM 79*, volume 72 of *Lecture Notes in Computer Science*, pages 216–226, 2007.