Notes on Boolean Function Analysis

Adrian She

March 2022

Abstract

This set of notes is intended as a summary of two lectures on Boolean function analysis given at University of Toronto's Theory Student Seminar. The goal is to introduce the basics of the subject and introduce its applications in computational social choice and approximation algorithms. Most of the results presented in these notes can be found in the textbook [O'D14], which is freely available online.

1 Introduction

A Boolean function f is a function $f : \{-1, 1\}^n \to \{-1, 1\}$ where $\{-1, 1\}^n$ is the set of n-bit strings. The study of Boolean functions is a central topic in theoretical computer science. Analysis of Boolean functions refers to a method of studying Boolean functions using Fourier analysis, or equivalently their representation as multilinear polynomials.

Definition 1. Given a subset $S \subseteq [n]$, the parity function is defined by $\chi_S(x_1, \dots, x_n) = \prod_{i \in S} x_i$.

The *Fourier expansion* of a Boolean function f is an expansion of f as a polynomial:

$$f(x_1, \dots, x_n) = \sum_{S \subseteq [n]} \hat{f}(S)\chi_S = \sum_{S \subseteq [n]} \hat{f}(S) \prod_{i \in S} x_i.$$

For instance, let Maj_n be the majority function on n bits, which outputs 1 if at least $\frac{n}{2}$ input bits are equal to 1. Then

$$Maj_3(x_1, x_2, x_3) = \frac{1}{2}x_1 + \frac{1}{2}x_2 + \frac{1}{2}x_3 - \frac{1}{2}x_1x_2x_3.$$

We record the following lemma which shows some properties of the Fourier expansion. We define an inner-product on the space of functions $f : \{-1, 1\}^n \to \mathbb{R}$ by setting $\langle f, g \rangle = \frac{1}{2^n} \sum_{x \in \{-1,1\}^n} f(x)g(x)$ given any two such functions f, g.

Lemma 1. *Let $\mathcal{F}$ be the vector space of all functions $f : \{-1, 1\}^n \to \mathbb{R}$. Then,*

1. *The set of 2^n parity functions χ_S form an orthonormal basis of $\mathcal{F}$ under the inner product. Therefore, the Fourier expansion of a Boolean function exists and is unique.*

2. *The Fourier coefficients of f can be computed by $\hat{f}(S) = \langle f, \chi_S \rangle$.*

3. *The Fourier coefficients satisfy $\sum_{S \subseteq [n]} \hat{f}(S)^2 = \langle f, f \rangle$.*

4. *The inner product satisfies $\langle f, g \rangle = \sum_{S \subseteq [n]} \hat{f}(S)\hat{g}(S)$ for any $f, g \in \mathcal{F}$.*

Proof. See [O'D14, Chapter 1.4]. □

The goal of analysis of Boolean functions is to deduce their properties from knowledge of their Fourier expansion. We will see several examples in the rest of these notes.

2 Basic Concepts in Boolean Function Analysis

2.1 Influence of Boolean Functions

We let X be a uniformly distributed random variable over $\{-1,1\}^n$.

Definition 2. Given a string $x \in \{-1,1\}^n$ and a coordinate i, let $x^{\oplus i}$ be x with the i^{th} bit negated. If $f(x^{\oplus i}) \neq f(x)$, then i is a *pivotal coordinate* for f at x.

For example, if $f = Maj_3$, then 1 is pivotal for f at $(-1,-1,1)$ since $f(-1,-1,1) = -1$ but $f(1,-1,1) = 1$.

Definition 3. The *influence* of coordinate i for f is the probability $\mathbf{Inf}_i(f) = \mathbb{P}[f(X^{\oplus i}) \neq f(X)]$. Equivalently it is the fraction of strings $x \in \{-1,1\}^n$ for which i is pivotal for f at x.

For example, $\mathbf{Inf}_1(Maj_3) = \frac{1}{2}$ since there are exactly 4 pivotal inputs: $(-1,-1,1), (1,-1,1), (1,1,-1), (-1,1,-1)$ out of 8.

Definition 4. Let x be a string, and let $x^{i\to b}$ be x with the i^{th} bit replaced by b. The i^{th} *discrete derivative* of f is the function defined by

$$D_i f(x) = \frac{f(x^{i\to 1}) - f(x^{i\to -1})}{2}.$$

Observe that $D_i f(x) = \begin{cases} 0 & i \text{ not pivotal for } f \text{ at } x \\ \pm 1 & i \text{ is pivotal for } f \text{ at } x \end{cases}.$

Hence, we conclude that $\mathbf{Inf}_i(f) = \mathbb{E}_{X\sim\{-1,1\}^n}[(D_i f(X))^2]$. This gives us a way to compute influences using the Fourier expansion.

Theorem 1. *Let $f = \sum_{S\subseteq[n]} \hat{f}(S)\chi_S$. Then:*

1. *The Fourier expansion of the function $D_i f$ is $D_i f = \sum_{S\subseteq[n] i\in S} \hat{f}(S)\chi_{S\setminus\{i\}}$.*
2. *The influence of coordinate i for f is equal to $\mathbf{Inf}_i(f) = \sum_{i\in S} \hat{f}(S)^2$.*

Proof. See [O'D14, Theorem 2.20]. □

Observe that this justifies the name "derivative" for the operator D_i, since D_i acts as partial differentiation operator $\frac{\partial}{\partial x_i}$ on the Fourier expansion.

The notion of influence can be applied in complexity theory, in particular to study the computational power of small-depth circuits (also known as $\mathbf{AC}^0$ circuits). See [O'D14, Chapter 4] for further details.

2.2 Noise Stability of Boolean Functions

Definition 5. Let $-1 \leq \rho \leq 1$. Random variables (X,Y) are *ρ-correlated* if X is chosen uniformly at random from $\{-1,1\}^n$, and the bits of Y are assigned as

$$Y_i = \begin{cases} X_i \text{ with probability } \frac{1}{2} + \frac{1}{2}\rho \\ -X_i \text{ with probability } \frac{1}{2} - \frac{1}{2}\rho \end{cases}.$$

Observe that if (X,Y) are ρ-correlated, then for each bit $\mathbb{E}[X_i] = \mathbb{E}[Y_i] = 0$, but $\mathbb{E}[X_iY_i] = \rho$.

Definition 6. Let (X,Y) be ρ-correlated. The *ρ-noise stability* of a function $f : \{-1,1\}^n \to \mathbb{R}$ is $\mathrm{Stab}_\rho(f) = \mathbb{E}_{(X,Y)}[f(X)f(Y)]$. In particular, if f has output ± 1, then $\mathrm{Stab}_\rho(f) = 2\mathbb{P}_{(X,Y)}[f(X) = f(Y)] - 1$.

Intuitively, functions with high noise stability change little under perturbations of their input. Conversely, the parity function $\prod_{i=1}^n x_i$ will have low noise stability since changing any input bit changes the output value.

We again have a formula for the noise stability in terms of the Fourier expansion.

Theorem 2. *Let $-1 \leq \rho \leq 1$ and suppose $f = \sum_{S \subseteq [n]} \hat{f}(S)\chi_S$. Then,*

1. *The noise stability of the parity function χ_S is $Stab_\rho(\chi_S) = \rho^{|S|}$.*
2. *The noise stability of f is $Stab_\rho(f) = \sum_{S \subseteq [n]} \hat{f}(S)^2 \rho^{|S|}$*

Proof. See [O'D14, Proposition 2.47]. □

3 Application 1: Social Choice

To see how the idea of noise stability can be applied, we consider a problem in social choice. We consider an election with 3 candidates a, b, c. Suppose in a certain society, a third of the population has preferences $a > b > c$, another third has preferences $c > a > b$ and another third has preferences $b > c > a$. Then aggregating the preferences, we see that a majority of the population prefer a over b, and a majority of the population prefer b over c. However, a majority of the population also prefers c over a! This non-transitivity of preferences is known as a *Condorcet paradox*, first observed by the Marquis de Condorcet in 1785 [DC14], and appeared to imply that the system of majority vote may lead to inconclusive results.

In the general set up, we fix some voting rule $f : \{-1, 1\}^n \to \{-1, 1\}$ that aggregates a result from n voters casting votes in a two candidate election. Given candidates a, b, c where each of n votes has a ranking over the candidates:

- We hold a two-candidate election with a (+1) and b (-1), votes x and outcome $f(x)$.
- We hold a two-candidate election with b (+1) and c (-1), votes y and outcome $f(y)$.
- We hold a two-candidate election with c (+1) and a (-1), votes z and outcome $f(z)$.

We say that an election has a Condorcet winner if the situation of cyclic preferences is avoided. This condition is sometimes known as *rationality*.

Definition 7. The election has a *Condorcet winner* if there is a candidate that wins all of the pair-wise elections that it participates in. In particular, for a three-candidate election, the election outcome $(f(x), f(y), f(z))$ is not $(1, 1, 1)$ or $(-1, -1, -1)$.

Using these definitions, we can compute the probability of a Condorcet winner in certain situations.

Theorem 3. *Suppose the voters come from an impartial culture, that is voters select a ranking of the three candidates uniformly at random. Then under voting rule f, the probability an election has a Condorcet winner is $\frac{3}{4}(1 - Stab_{-1/3}(f))$.*

Proof. Let $NAE_3 : \{-1, 1\}^3 \to \{0, 1\}$ be the function $NAE_3(x, y, z) = \begin{cases} 0 & x = y = z \\ 1 & \text{otherwise} \end{cases}$. Then

$$NAE_3(x, y, z) = \frac{3}{4} - \frac{1}{4}xy - \frac{1}{4}xz - \frac{1}{4}yz,$$

and the probability that the election has a Condorcet winner is $\mathbb{E}_{x,y,z}[NAE_3(f(x), f(y), f(z))]$ where x, y, z are the votes cast. Each vote (x_i, y_i, z_i) is chosen uniformly at random from the set where $NAE_3(x, y, z) = 1$:

$$\{(-1, 1, 1); (1, -1, 1); (1, 1, -1); (1, -1, -1); (-1, 1, -1); (-1, -1, 1)\},$$

so we see that $\mathbb{E}[x_i] = \mathbb{E}[y_i] = \mathbb{E}[z_i] = 0$ and $\mathbb{E}[x_i y_i] = \mathbb{E}[y_i z_i] = \mathbb{E}[x_i z_i] = \frac{2}{6} - \frac{4}{6} = -\frac{1}{3}$. Hence using the definition of noise stability, we have

$$\mathbb{E}[NAE_3(f(x), f(y), f(z))] = \frac{3}{4} - \frac{3}{4}\text{Stab}_{-1/3}(f).$$

□

As a corollary of this calculation, we obtain a proof of Arrow's Impossibility Theorem [Arr50], which places a strong condition on all voting rules that avoid the Condorcet paradox. This proof was originally due to Kalai [Kal02]. A proof is also available at [Tal20] or [O'D14, Section 2.5].

Theorem 4. *Suppose f is a unanimous voting rule. (i.e. if all votes are for candidate i then i should be selected). If there is always a Condorcet winner in a three-candidate election, then f is a dictatorship function (eg. $f(x) = x_i$ for some index i).*

Proof. If there is always a Condorcet winner, then

$$1 = \frac{3}{4} - \frac{3}{4}\text{Stab}_{-1/3}(f) = \frac{3}{4} - \frac{3}{4}\sum_{S\subseteq[n]}(-\frac{1}{3})^{|S|}\hat{f}(S)^2 = \frac{3}{4} - \frac{3}{4}\sum_{k=0}^{n}(-\frac{1}{3})^k W^k(f),$$

where $W^k(f) = \sum_{S\subseteq[n]|S|=k}\hat{f}(S)^2$. This is equivalent to $\sum_{k=0}^{n}(-\frac{1}{3})^k W^k(f) = -\frac{1}{3}$. Observe that

$$\sum_{k=0}^{n}(-\frac{1}{3})^k W^k(f) \geq -\frac{1}{3}W^1(f) - \frac{1}{27}(1 - W^1(f)),$$

so $\sum_{k=0}^{n}(-\frac{1}{3})^k W^k(f) = -\frac{1}{3}$ implies that $W^1(f) = 1$.

Observe that all functions with $W^1(f) = 1$ satisfy $f(x) = x_i$ or $f(x) = -x_i$ for some index i. To see this, we write $f(x) = \sum_{i=1}^{n} a_i x_i$, and conclude that the derivative $D_j f(x) = a_j$ is non-zero and constant for some index j. However, since f is Boolean, $D_j f(x) \in \{-1, 0, 1\}$ for each $x \in \{-1, 1\}^n$ Hence, $a_j = \pm 1$ and the condition $\sum_{i=1}^{n} a_i^2 = 1$ implies that $f = \pm x_j$ for some index j.

Since the voting rule is unianimous, we conclude that our voting rule is $f(x) = x_j$ for some index j. □

3.1 The Noise Stability of Majority

Using the formula computed in Theorem 3, we may also wonder about the probability of a Condorcet winner using commonly used voting rules (eg. when f is a majority vote). The motivates the computation of the noise stability of majority. We can get a good approximation of the noise stability using the central limit theorem, as described in [O'D14, Section 5.2].

Let $\text{sgn}(x) = \begin{cases} 1 & x > 0 \\ -1 & x < 0 \end{cases}$ be the sign function, and we observe that the majority function on n bits can be rewritten as $Maj_n(x) = \text{sgn}(\sum_{i=1}^{n}\frac{1}{\sqrt{n}}x_i)$.

Observe that if (x, y) are ρ-correlated string of length ns, then by the central limit theorem, the vector $(\frac{1}{\sqrt{n}}\sum_{i=1}^{n} x_i, \frac{1}{\sqrt{n}}\sum_{i=1}^{n} y_i)$ converges in distribution to a normal random variables (S_x, S_y) with mean $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and covariance matrix $\begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$ as $n \to \infty$. Therefore, as $n \to \infty$, we can approximate

$$\text{Stab}_\rho(Maj_n) = \mathbb{E}_{(X,Y)}[Maj_n(X)Maj_n(Y)] \to \mathbb{E}[\text{sgn}(S_x)\text{sgn}(S_y)] = 1 - 2\mathbb{P}(S_x S_y < 0).$$

Observe that we can generate (S_x, S_y) by picking any two unit vectors $u, v \in \mathbb{R}^2$ with inner product $\langle u, v\rangle = \rho$, and setting $S_x = \langle u, g\rangle$ and $S_y = \langle v, g\rangle$ for some normal random vector g with mean $\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ and covariance $\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Now observe that $S_x S_y < 0$ exactly when $\langle u, g\rangle < 0$ or $\langle v, g\rangle > 0$ or vice versa. In other words, the angle between u and g is acute or the angle between v and g is obtuse (or vice versa). This happens exactly the line g' perpendicular to g cuts the angle between u, v. By rotational symmetry of g and g', we conclude that

$$\mathbb{P}(S_x S_y < 0) = \frac{\theta}{\pi},$$

where $\theta = \arccos\rho$ is the angle between u and v. Therefore, we conclude:

Theorem 5. *The ρ-noise stability of the majority function Maj_n approaches $1 - \frac{2 \arccos \rho}{\pi}$ as $n \to \infty$.*

Hence, using Theorem 3, we can also conclude:

Theorem 6 (Guilbaud's Formula)**.** *In an three candidate election using the majority vote rule where voters have impartial preferences, the probability of a Condorcet winner approaches $\frac{3}{2\pi} \arccos(-\frac{1}{3}) = 0.912...$ as the number of voters tends to infinity.*

4 Application 2: Property Testing and CSPs

4.1 Property Testing

A second application of Boolean function analysis involves the theory of approximation algorithms and constraint satisfaction problems. To do this we first need to introduce the idea of property testing. A property $\mathcal{P}$ is a subset of all Boolean functions. One example we will consider is linearity testing: the set of all Boolean functions satisfying $f(x+y) = f(x) + f(y)$ (if $f : \{0,1\}^n \to \{0,1\}$) .

Definition 8. Given two Boolean functions f, g, the *distance* between them is $\text{dist}(f,g) = \mathbb{P}[f(x) \neq g(x)]$, which is the probability they differ on a uniform random input $x \in \{-1,1\}^n$.

Given a property $\mathcal{P}$ and a function f, the distance between f and $\mathcal{P}$ is $\text{dist}(f, \mathcal{P}) = \min_{g \in \mathcal{P}} \text{dist}(f,g)$.

Definition 9. An *r-query property tester* for $\mathcal{P}$ aims to decide whether or not a given function f belongs to $\mathcal{P}$ or not using the following procedure:

- Chooses strings $x_1, \ldots, x_r \in \{-1,1\}^n$ according to some distribution.
- Evaluates $f(x_1), \ldots, f(x_r)$.
- Based on the outcomes, decide if $f \in \mathcal{P}$ or not (deterministically)

We require that if $f \in \mathcal{P}$, then the test accepts f with probability one. Otherwise, there should be some constant λ such that if $\text{dist}(f, \mathcal{P}) \geq \epsilon$, then the test accepts with probability $\leq 1 - \lambda\epsilon$.

A natural goal in property testing is to minimize the number of queries needed. Here, we will present a three-query test for linearity, developed by BLR in [BLR93]. The test proceeds as follows:

- Choose strings x, y uniformly at random and compute $x + y$.
- Check if $f(x) + f(y) = f(x+y)$. If this condition is satisfied, accept, otherwise reject.

Theorem 7. *If the BLR linearity test accepts function f with probability at least $1 - \epsilon$, then dist$(f, \chi_S) \leq \epsilon$ for some linear function χ_S.*

Proof. See [O'D14, Theorem 1.30]. □

4.2 Constraint Satisfaction Problems (CSPs)

There is a close connection between the theory of property testing and constraint satisfaction problems. We first give a few examples of CSPs before stating the formal definition:

- In **Max-3-SAT**, one is given a Boolean formula in conjunctive normal form, and one finds to find an assignment to the variables that satisfies as many clauses as possible.
- In **Max-Cut**, one is given a graph $G = (V, E)$, and one wants to find a partitioning of the vertex set such that as many edges cross the cut as possible.

Both of these problems can be phrased as versions of the Max-CSP problem we will now formally describe.

Definition 10. A *CSP* consists of a domain Ω and a set of predicates Ψ, where each $\psi \in \Psi$ is a function $\psi : \Omega^r \to \{0,1\}$. We call r the *arity* of the constraint ψ.

An instance of Max-CSP(Ψ) over a set of variables V consists of a list $(S = \{v_1, \ldots, v_r\}, \psi)$ where each $S \subseteq V$ is a subset of variables of the appropriate arity for $\psi \in \Psi$. A constraint is satisfied by assignment $F : V \to \Omega$ if $\psi(F(v_1), \ldots, F(v_r)) = 1$.

The value $val(I)$ of an instance I is the maximum fraction of satisfied constraints over all possible assignments $F : V \to \Omega$. The goal of the Max-CSP(Ψ) problem is to find an assignment $F : V \to \Omega$, which maximizes the $val(I)$, given any instance of the problem.

For the two CSPs (**Max-3-Sat** and **Max-Cut**) we have described, it is **NP**-Hard to decide if an instance I of a CSP is satisfiable (eg. distinguishing instances where $val(I) = 1$ versus the case where $val(I) < 1$). This motivates the study of *approximation algorithms* for CSPs.

Definition 11. Let $0 \le \alpha < \beta \le 1$. An algorithm is an (α, β)-approximation algorithm for Max-CSP(Ψ) if on any instance I with $val(I) \ge \beta$, the algorithm outputs an assignment with $val(I) \ge \alpha$.

For example, the Goemans-Williamson algorithm for **Max-Cut** is polynomial time approximation algorithm achieving an $(0.878\beta, \beta)$-approximation. The algorithm is briefly described in [O'D14, Section 11.7].

However, assuming Khot's Unique Games Conjecture and existence of an appropriate property tester, there are strong limitations on polynomial-time approximation algorithms for CSPs.

Definition 12. The *ρ-stable influence* of i on f is

$$\mathbf{Inf}_i^{\rho}(f) = \mathrm{Stab}_\rho(D_i f) = \sum_{i \in S} \rho^{|S|-1} \hat{f}(S)^2.$$

Definition 13. Let Ψ be a set of predicates over the domain $\{-1,1\}$, and $0 \le a < \beta \le 1$. A (α, β)-*dictator versus no notables test* using predicate set Ψ is a property test satisfying the following conditions:

- If f is a dictator function (eg. $f(x) = x_i$ for some index i), then the test accepts with probability at least β.
- If f is "far" from being a dictator function, in the sense that if has no (ϵ, ϵ)-notable coordinates (every coordinate satisfies $\mathbf{Inf}_i^{1-\epsilon}(f) \le \epsilon$), then the test accepts with probability at most $\alpha + \lambda(\epsilon)$ for some function $\lambda(\epsilon) \to 0$ as $\epsilon \to 0$.
- The acceptance or rejection decision of the tester uses evaluations of the function f and predicates from Ψ only.

The following is the key result connecting property testing and CSPs, assuming the unique games conjecture, first proven in [KKMO07].

Theorem 8. *Assume the Unique Games Conjecture. Suppose we have a CSP over domain $\Omega = \{-1,1\}$ and predicate set Ψ. If there is an (α, β)-dictator versus no notables test using predicate set Ψ, then for every $\delta > 0$, it is* **NP***-hard to $(\alpha + \delta, \beta - \delta)$-approximate Max-CSP($\Psi$).*

As an application of this theorem, we consider the hardness of approximating the CSP **Max-3-Lin**. In this CSP, the domain is $\{0,1\}$ and constraints are linear equations of the form $x_i + x_j + x_k = b$ where $b \in \{0,1\}$. Note that we have an approximation algorithm for **Max-3-Lin** with approximation factor $\frac{1}{2}$ since a random assignment of the variables to $\{0,1\}$ satisfies half of the equations in expectation. However, we have the following result:

Theorem 9. *Assume the Unique Games Conjecture, then it is* **NP***-hard to approximate Max-3-Lin with a factor of $(\frac{1}{2} + \delta, 1 - \delta)$ for any $\delta > 0$.*

By our theorem, it suffices to design a $(\frac{1}{2}, 1-\delta)$ dictator versus no notables test whose decision predicates are of the form $f(x) + f(y) + f(z) = b$ (or $f(x)f(y)f(z) = b$ over the $\{-1,1\}$ basis) to prove the hardness result. It turns out that appropriately modifying the BLR test produces the needed property test with the given parameters. The Hastad test performs the following steps given query access to a function $f : \{-1,1\}^n \to \{-1,1\}$:

- Choose $x, y \in \{-1, 1\}^n$ uniformly and independently.
- Choose a bit $b \in \{-1, 1\}$ uniformly and let z be the string defined bitwise by $z_i = b(x_i y_i)$.
- Let z' be z where each bit is flipped with probability $1 - \delta$.
- Accept if $f(x)f(y)f(z') = b$.

Theorem 10. *The Hastad test is a* $(\frac{1}{2}, 1 - \frac{\delta}{2})$ *dictator-versus-no notables test, using the predicate set from Max-3-Lin.*

Proof. See [O'D14, Section 7.4] for details. □

A property test construction can be used to prove optimality of the Goemans-Williamson algorithm for Max-Cut assuming the unique games conjecture and the "majority is stablest" theorem proved in [MOO05], as discussed in [O'D14, Section 11.7].

4.3 Open Problems about Approximation Algorithms

Finally, we note that there are still many open problems concerning approximability of CSPs. One interesting open problem concerns quantum CSPs. In classical CSPs, the goal is to find a bit string $\{0, 1\}^n$ that maximizes a set of given constraints. For quantum CSPs, the analogous problem is to find a complex vector $|\psi\rangle$ of dimension 2^n and unit norm that maximizes the inner product $\langle\psi|H|\psi\rangle$ for some Hermitian matrix H. The matrix H is known as the Hamiltonian. Equivalently, one wants to compute a good approximation to the maximum eigenvalue of H. Writing down all components of a vector of dimension 2^n or the matrix H requires exponential time, so one usually restricts attention to vectors and matrices with a polynomial time description.

For example, one can maximize over the set of product state vectors, which are vectors where $|\psi\rangle$ is expressible as a tensor product $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_n\rangle$ with each $|\psi_i\rangle \in \mathbb{C}^2$. Since we now have a polynomial size description of a product state $|\psi\rangle$ (ie. one only needs to write down $2n$ numbers rather than 2^n to describe the vector), we can investigate if one can efficiently find the optimal product state solution for the quantum CSP in polynomial time. This assumption that the maximizer of $\langle\psi|H|\psi\rangle$ is given by a product state is not always satisfied, but in many cases of interest the optimal state is well-approximated by product states. For a rigorous analysis for when product states are a good approximation, see [BH13].

In recent work by [HNP$^+$21], it is proven that under the unique games conjecture it is hard to find good product state approximations for a certain class of Hamiltonians. We recall that the Pauli matrices are the 2 by 2 matrices given by

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \quad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Tensor products of the Pauli matrices are then operators on a vector space with larger dimension.

Given a graph G with n vertices, we define the Max-Cut Hamiltonian by

$$H_{MC} = \sum_{(i,j) \in E} \frac{I \otimes I - Z_i \otimes Z_j}{2},$$

and the Quantum Max-Cut Hamiltonian by

$$H_{QMC} = \sum_{(i,j) \in E} \frac{I \otimes I - X_i \otimes X_j - Y_i \otimes Y_j - Z_i \otimes Z_j}{2}.$$

We observe that the name max-cut Hamiltonian is justified since the maximum eigenvalue of H_{MC} is equal to the number of edges in a max cut of G. Much less is known about the quantum max-cut problem, compared to the classical max-cut problem where we have an optimal polynomial time algorithm (at least assuming the unique games conjecture). The following results are the current state of the art on the quantum max-cut problem:

Theorem 11. *Let G be a graph and H_{QMC} be the quantum max-cut Hamiltonian. Recall that our objective is to find a state of unit norm $|\psi\rangle$ that maximizes $\langle\psi|H_{QMC}|\psi\rangle$.*

1. *It is possible to output a state $|\psi\rangle$ which is a tensor product of one and two qubit states in randomized polynomial time, achieving an approximation factor of at least 0.53. [AGM20, PT21]*
2. *Assuming the Unique Games Conjecture, it is* **NP***-hard to compute a $(0.956+\epsilon)$-approximation to the optimal state, or optimal product state, any $\epsilon > 0$. [HNP+21]*

The techniques in [HNP+21] use many of the notions of Fourier analysis introduced in these notes, such as noise stability of functions. These notions in their paper are also generalized from Boolean functions to real-valued functions. It is an interesting open problem to improve either of the results mentioned in Theorem 11 for quantum max-cut.

References

[AGM20] Anurag Anshu, David Gosset, and Karen Morenz. Beyond product state approximations for a quantum analogue of max cut. *arXiv preprint arXiv:2003.14394*, 2020.

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

[BH13] Fernando GSL Brandao and Aram W Harrow. Product-state approximations to quantum ground states. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 871–880, 2013.

[BLR93] Manuel Blum, Michael Luby, and Ronitt Rubinfeld. Self-testing/correcting with applications to numerical problems. *Journal of Computer and System Sciences*, 47(3):549–595, 1993.

[DC14] Nicolas De Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. Cambridge University Press, 2014.

[HNP+21] Yeongwoo Hwang, Joe Neeman, Ojas Parekh, Kevin Thompson, and John Wright. Unique games hardness of quantum max-cut, and a vector-valued Borell's inequality. *arXiv preprint arXiv:2111.01254*, 2021.

[Kal02] Gil Kalai. A Fourier-theoretic perspective on the Condorcet paradox and Arrow's theorem. *Advances in Applied Mathematics*, 29(3):412–426, 2002.

[KKMO07] Subhash Khot, Guy Kindler, Elchanan Mossel, and Ryan O'Donnell. Optimal inapproximability results for max-cut and other 2-variable CSPs? *SIAM J. Comput.*, 37(1):319–357, apr 2007.

[MOO05] Elchanan Mossel, Ryan O'Donnell, and Krzysztof Oleszkiewicz. Noise stability of functions with low influences: invariance and optimality. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pages 21–30. IEEE, 2005.

[O'D14] Ryan O'Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

[PT21] Ojas Parekh and Kevin Thompson. Application of the level-2 quantum Lasserre hierarchy in quantum approximation algorithms. *arXiv preprint arXiv:2105.05698*, 2021.

[Tal20] Avishay Tal. Analysis of boolean function course notes, 2020. `https://www.avishaytal.org/cs294-analysis-of-boolean-functions`.