These lecture notes are based on a manuscript of a book on matrix rigidity by Chi-Ning Chou and Sasha Golovnev.

Chapter 1

Introduction

1.1 Definitions and examples

Lecture 1

One of the main questions in computational complexity is that of proving lower bounds on the size of Boolean circuits computing explicitly given functions. While most Boolean functions of n inputs require circuits of size $2^n/n$ [Sha49a, Lup59a], we can only prove small linear lower bounds for explicitly defined functions [LR01, IM02, Blu83, DK11, FGHK16]. [1]

The same question remains open for *linear* circuits computing *linear* Boolean functions. Since any linear function with one output can be computed by a circuit of size at most n, we study linear functions with n inputs and n outputs. A random linear map with n outputs requires circuits of size $n^2/\log n$ [Lup56], but the best known lower bound for an explicit linear map is only $3n - o(n)$ [Cha94a].

The notion of matrix rigidity was introduced by Valiant [Val77] as a tool for proving lower bounds against linear circuits. (A related notion of separability was introduced by Grigoriev [Gri76].)

We will use the following notation. A matrix A is called s-sparse, if the number of non-zero entries in A is at most s. We will use $I_n, 0_n$ and J_n to denote the identity matrix, zero matrix, and all-ones matrix of size $n \times n$. For a matrix $A \in \mathbb{F}^{n\times n}$, by $\|A\|_0$ we denote the number of non-zero entries in A.

Definition 1.1 (Rigidity)**.** *Let $\mathbb{F}$ be a field, $A \in \mathbb{F}^{n\times n}$ be a matrix, and $0 \le r \le n$. The rigidity of A over $\mathbb{F}$, denoted by $\mathcal{R}_A^{\mathbb{F}}(r)$, is the Hamming distance between A and the set of matrices of rank at most r. Formally,*

$$\mathcal{R}_A^{\mathbb{F}}(r) := \min_{\mathrm{rank}(A+C)\le r} \|C\|_0 \ .$$

In other words, a matrix A has rigidity $\mathcal{R}_A^{\mathbb{F}}(r) \ge s$ if and only if $A \in \mathbb{F}^{n\times n}$ *cannot* be written as a sum

$$A = S + L \ ,$$

where $S \in \mathbb{F}^{n\times n}$ is $(s-1)$-sparse matrix, and $L \in \mathbb{F}^{n\times n}$ is low-rank: $\mathrm{rank}(L) \le r$.

Valiant [Val77] proved that any linear map $A \in \mathbb{F}^{n\times n}$ computed by a linear circuit (over a field $\mathbb{F}$) of depth $O(\log n)$ and size $o(n \log\log n)$ has rigidity at most $\mathcal{R}_A^{\mathbb{F}}(\varepsilon n) \le n^{1+\delta}$ for every constant $\varepsilon, \delta > 0$. Therefore, an explicit matrix of higher rigidity would give us a super-linear lower bound against linear circuits of logarithmic depth. Despite more than 40 years of research, the problem of proving super-linear lower bounds for such circuits remains open.

Let us now see the rigidity of a few specific matrices.

- If $A \in \mathbb{F}^{n\times n}$ has rank $\mathrm{rank}(A) = k$ over the field $\mathbb{F}$, then $\mathcal{R}_A^{\mathbb{F}}(r) = 0$ for every $r \ge k$. Indeed, A can be written as a sum of A and 0_n, where $\mathrm{rank}(A) \le r$ and 0_n is 0-sparse. Similarly, an s-sparse matrix $A \in \mathbb{F}^{n\times n}$ has rigidity $\mathcal{R}_A^{\mathbb{F}}(r) \le s$ for any value of r.

[1] Here by explicit functions we mean functions computable in time polynomial in n. We will later discuss the notion of explicitness in greater detail.

- For any $0 \le r \le n$, $\mathcal{R}^{\mathbb{F}}_{I_n}(r) = n - r$. Indeed, if we change $n-r$ ones of I_n to zeros, then the resulting matrix has rank r, which implies that $\mathcal{R}^{\mathbb{F}}_{I_n}(r) \le n-r$. On the other hand, for any $(n-r)$-sparse matrix B, from subadditivity of rank,

$$\operatorname{rank}(I_n + B) \ge \operatorname{rank}(I_n) - \operatorname{rank}(B) \ge n - (n-r) = r \ ,$$

  which gives us that $\mathcal{R}^{\mathbb{F}}_{I_n}(r) \ge n - r$.

- Let n be a multiple of $2r$, and let $M_n \in \mathbb{F}^{n \times n}$ be a matrix consisting of matrices I_{2r} stacked together side by side:

$$M_n = \begin{pmatrix} I_{2r} & \cdots & I_{2r} \\ \vdots & \ddots & \vdots \\ I_{2r} & \cdots & I_{2r} \end{pmatrix} \ .$$

  We will show that this matrix has rigidity $\mathcal{R}^{\mathbb{F}}_A(r) = \frac{n^2}{4r}$.

  Theorem 1.2 ([Mid05])**.** *For any field $\mathbb{F}$, and any n divisible by $1 \le 2r \le n$,*

$$\mathcal{R}^{\mathbb{F}}_{M_n}(r) = \frac{n^2}{4r} \ .$$

  Proof of Theorem 1.2. M_n consists of $\frac{n^2}{4r^2}$ copies of the identity matrix I_{2r}. In order to drop the rank of A to r, the rank of each copy of I_{2r} must be dropped to r. From the previous example we know that in order to decrease the rank of I_{2r} to r, one needs to change at least r elements. Thus, $\frac{n^2}{4r^2} \cdot r = \frac{n^2}{4r}$ entries of M_n must be changed. Note that this bounds is tight, *i.e.*, $\mathcal{R}^{\mathbb{F}}_{M_n}(r) = \frac{n^2}{4r}$. □

The bound of Theorem 1.2 easily generalizes to all values $r \le n/2$ with a loss of a multiplicative factor of 2. This theorem was proven by Midrijānis [Mid05], and it gives a simple matrix with rigidity $\mathcal{R}^{\mathbb{F}}_{M_n}(r) \ge \frac{n^2}{8r}$.

We will see later that there exist matrices with much higher rigidity $\tilde{\Omega}\left((n-r)^2\right)$. Embarrassingly, the best known lower bound for an *explicit* matrix improves on the $\frac{n^2}{8r}$ bound only by a logarithmic factor.

1.2 Circuit Complexity

A circuit corresponds to a simple straight line program where every instruction performs a binary operation on two operands, each of which is either an input or the result of a previous instruction. The structure of this program is extremely simple: no loops, no conditional statements. Still, we know no functions in P (or even NP, or even E^{NP}) that requires even $3.1n$ binary instructions ("size") to compute on inputs of length n. This is in sharp contrast with the fact that it is easy to *non-constructively* find such functions: simple counting arguments show a random function on n variables has circuit size $\Omega(2^n/n)$ with probability $1 - o(1)$ [Sha49b, Lup59b].

For small-depth circuits we know several strong lower bounds. (Note that when working with circuits of constant depth, we do not pose bounds on the fan-ins of the gates.) Depth-2 circuits (after a simple normalization) are just CNFs or DNFs. It is easy to see that the parity function $\oplus_n$ of n inputs requires CNFs and DNFs of size $\Omega(2^n)$. For depth-d circuits, we know a lower bound of $2^{\Omega(n^{(1/(d-1))})}$ [Hås86, HJP93, PPZ97, Bop97, PPSZ05, MW17]. Thus, for depth $d = o(\log n / \log\log n)$ we have non-trivial lower bounds even if the fan-ins of the gates are unbounded. For circuits with fan-in 2, we known functions which cannot be computed by circuits of depth $1.99 \log n$ [Nec66]. Thus, a problem on the frontier is

Problem 1.3. *Prove a lower bound of $10n$ against circuits of depth $10 \log n$.*
More generally, a lower bound of $\omega(n)$ against circuits of depth $O(\log n)$.

Super-linear lower bounds are not known even for linear circuits, i.e., circuits consisting of only gates computing linear combinations of their two inputs. Note that every linear function with one output has a circuit of size $n-1$ (and depth $\log n$). For linear circuits, we consider *linear transformations*, multi-output

functions of the form $f(x) = Ax$ where $A \in \mathbb{F}^{n\times n}$. For a random matrix $A \in \{0,1\}^{n\times n}$, the size of the smallest linear circuit computing Ax is $\Theta(n^2/\log n)$ [Lup56] with probability $1 - o(1)$, but for explicitly-constructed matrices the strongest known lower bound is $3n - o(n)$ [Cha94b]. This leads us to another problem on the frontier:

Problem 1.4. *Prove a lower bound of $\omega(n)$ against* linear *circuits of depth $O(\log n)$.*

Formally, Problem 1.3 and Problem 1.4 are incomparable, as in the linear case we study a weaker computational model (which makes it easier to prove lower bounds), but are limited to proving lower bounds for a smaller class of problem (which makes it harder to prove lower bounds).

1.3 Circuits and Rigidity

In this section, we will present a seminal result of Valiant [Val77] showing that rigid matrices require log-depth circuits of super-linear size. We start with the definition of linear circuits.

Definition 1.5 (Linear circuits)**.** *Let $\mathbb{F}$ be a field and $n \in \mathbb{N}$. A circuit C with n inputs and n outputs is a directed acyclic graph where n vertices have fan-in zero and are labeled by the inputs, all other vertices have fan-in two and are labeled with* affine functions *(over $\mathbb{F}$) of their two inputs, n of these vertices are labeled as outputs. For every fixed input, the value at each node is computed by applying the corresponding functions. Such a circuit C naturally defines a linear map $f\colon \mathbb{F}^n \to \mathbb{F}^n$, and the corresponding matrix $A \in \mathbb{F}^{n\times n}$ such that $f(x) = Ax$.*

The depth $d(C)$ of a circuit C is the length of the longest path in the circuit. The size $s(C)$ of C is defined as the number of vertices in C.

The following theorem shows a connection between lower bounds for linear circuits and matrix rigidity.

Theorem 1.6. *Let $\mathbb{F}$ be a field, and $A \in \mathbb{F}^{n\times n}$ be a family of matrices for $n \in \mathbb{N}$. If $\mathcal{R}_A^{\mathbb{F}}(\varepsilon n) > n^{1+\delta}$ for constant $\varepsilon, \delta > 0$, then any $O(\log n)$-depth linear circuit computing $x \to Ax$ must be of size $\Omega(n \cdot \log\log n)$.*

The proof of Theorem 1.6 repeatedly uses the following beautiful graph theoretic lemma due to Erdös, Graham, and Szemerédi [EGS76]: If G is a directed acyclic graph with s edges and of depth d, then there is a set of $s/\log d$ edges whose removal decreases the depth of G by a factor of two. We will follow the proof of this lemma from [Vio09].

Lemma 1.7 ([EGS76])**.** *Let G be an acyclic digraph with s edges and of depth $d = 2^k$. There exists a set of $s/\log d$ edges in G such that after their removal, the longest path in G has length at most $d/2$.*

Proof of Lemma 1.7. For ease of exposition, we follow [Vio09] and define a *depth function*. Let $G = (V, E)$ be an acyclic digraph. We say that $D : V \to \{0, 1, \ldots, d\}$ is a depth function for G if for any $(a, b) \in E$, $D(a) < D(b)$. It is not difficult to see that G has depth at most d if and only if there exists a depth function $D : V \to \{0, 1, \ldots, d-1\}$ for G.

We start with G of depth at most $d = 2^k$, and its depth function $D : V \to \{0, 1, \ldots, 2^k\}$. Now, consider the following partition of E using the depth function D. For each $i \in [k]$, define

$$E_i = \{(a, b) \in E : \text{ the most significant bit where } D(a), D(b) \text{ differ is the } i^{\text{th}} \text{ bit}\}.$$

As $\{E_i\}_{i\in[k]}$ is a partition of E, by the averaging argument, there exists $i^* \in [k]$ such that

$$|E_{i^*}| \le \frac{|E|}{k} \le \frac{|E|}{\log d}.$$

Now, it suffices to show that the depth of $G' = (V, E')$, where $E' = E \backslash E_{i^*}$, is at most 2^{k-1}. This can be shown by exhibiting a depth function $D' : V \to \{0, 1, \ldots, 2^{k-1} - 1\}$ for G'. The following shows that we can take $D'(v)$ to be $D(v)$ without the $i^{*\text{th}}$ bit.

Consider an edge $(a, b) \in E'$. Since $(a, b) \in E$, $D(a) < D(b)$. In particular, there exists $i \in [k]$ such that the most significant bit where $D(a)$ and $D(b)$ differ is i. Since $(a, b) \in E'$, the edge (a, b) was not removed, so $i \neq i^*$. Therefore, after removing the bit i^*, this bit i is still the most significant bit where $D'(a)$ and $D'(b)$ differ. This implies that $D'(a) < D'(b)$, and that $D' : V \to \{0, 1, \ldots, 2^{k-1} - 1\}$ is a depth function for G'. □

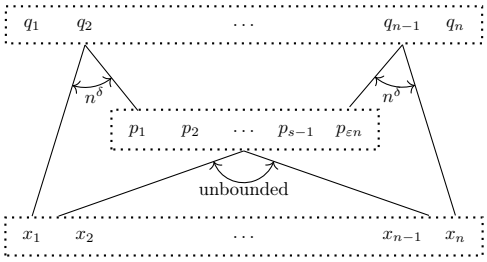

Figure 1.1: In order to compute the values of the outputs of the circuit C, first we precompute the values of εn removed edges (or vertices V'), and store them in variables p_i. Now each output q_j of the circuit C can be computed from n^δ inputs and precomputed bits. In particular, $A = BM + C$, where C encodes the dependence of the outputs q_i on the inputs x_j; B encodes the dependence of q_i on p_j; M encodes the dependence of p_i on x_j. Since C is sparse, and BM is low rank, the matrix A is not rigid.

Now we finish the proof of Theorem 1.6. **Lecture 2**

Proof of Theorem 1.6. We will show that for every constant $c_d \geq 2$, every circuit of depth at most $c_d \log n$ computing $x \to Ax$ must be of size at least $c_s n \log\log n$ for a constant $c_s = \frac{\varepsilon}{\log c_d + \log 1/\delta}$. Suppose, to the contrary, that there is a linear circuit C of size $s = c_s n \log\log n$ and depth $d = c_d \log n = 2^k$ that computes $x \to Ax$. Let G be the underlying acyclic digraph of C.

First, we apply Lemma 1.7 to G t times, and get a graph G' such that (i) only

$$s \cdot \left(\frac{1}{\log d} + \frac{1}{\log d - 1} + \cdots \frac{1}{\log d - (t-1)} \right) \leq \frac{st}{\log d - (t-1)}$$

edges are removed from G and (ii) the longest path in G' is of length at most $d' \leq d/2^t$.

By setting $t = \log c_d + \log 1/\delta$, the longest path in G' has length $\leq d/2^t = \delta \log n$, and the number of removed edges is at most

$$\frac{st}{\log d - (t-1)} = \frac{st}{\log d/2} \leq \frac{t c_s n \log\log n}{\log\log n} = \varepsilon n \,.$$

Now, let E be the set of removed edges and V' be the set of *tail* vertices of the edges from E. Since all paths in G' are no longer than d' and all in-degrees are at most 2, every output vertex in G' is now connected to at most $2^{d'}$ input variables. Therefore, every output is a (linear) function of at most $2^{d'}$ inputs and the functions computed at the removed edges (or the vertices V').

More specifically, let A_i be the i^{th} row of A, *i.e.,* the linear form computed by the i^{th} output vertex of G. Then A_i can be written as the following sum

$$A_i = \sum_{j \in [|V'|]} b_{ij} v_j + c_i$$

where v_j is the linear form computed by the j^{th} element in V' and c_i is the linear form computed by the i^{th} output vertex in G'. Note that since c_i only depends on at most $2^{d'}$ input variables.

Therefore, the matrix A can be written as follows.

$$A = BM + C$$

where $B \in \mathbb{F}^{n \times |V'|}$ consists of the coefficients b_{ij}, rows of $M \in \mathbb{F}^{|V'| \times n}$ compute linear forms of vertices from V', and $C \in \mathbb{F}^{n \times n}$ is a row sparse matrix where the number of non-zero entries in each row is at most $2^{d'} = n^\delta$.

The above argument gives us that $\tilde{\mathcal{R}}^{\mathbb{F}}_A(|V'|) = \tilde{\mathcal{R}}^{\mathbb{F}}_A(\varepsilon n) \leq n^\delta$, which contradicts the assumption on the rigidity of A.

□

1.4 Existence of Rigid Matrices

In this section, we will show that for any field $\mathbb{F}$, most of the $n \times n$ matrices have the highest possible rigidity for any rank parameter r.

It turns out that for every matrix A and field $\mathbb{F}$, there is a simple upper bound $\mathcal{R}_A^{\mathbb{F}}(r) \leq (n-r)^2$. Valiant [Val77] showed that this upper bound is essentially tight for a random matrix. First, we give a proof of the upper bound.

Theorem 1.8 (Simple upper bound)**.** *For any field $\mathbb{F}$, matrix $A \in \mathbb{F}^{n \times n}$, and integer $0 \leq r \leq n$, we have that*

$$\mathcal{R}_A^{\mathbb{F}}(r) \leq (n-r)^2 \ .$$

Proof of Theorem 1.8. If $\mathrm{rank}(A) \leq r$, then $\mathcal{R}_A^{\mathbb{F}}(r) = 0 \leq (n-r)^2$. Thus, it suffices to focus on the case where there is an $r \times r$ *full-rank* submatrix $B \in \mathbb{F}^{r \times r}$ of A. Without loss of generality, assume that B is located in the top left corner of A:

$$A = \begin{pmatrix} B & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \ , \tag{1.9}$$

where $A_{12} \in \mathbb{F}^{r \times (n-r)}, A_{21} \in \mathbb{F}^{(n-r) \times r}, A_{22} \in \mathbb{F}^{(n-r) \times (n-r)}$. In order to prove that $\mathcal{R}_A^{\mathbb{F}}(r) \leq (n-r)^2$, we will show that it is possible to change the entries in $A_{22} \in \mathbb{F}^{(n-r) \times (n-r)}$ and reduce the rank of A to r. Since B has full rank, each row in A_{21} is a *unique linear combination* of the rows in B. Thus, we can change the entries in A_{22} according to these linear combinations so that each row in A is now a linear combination of the first r rows, *i.e.,* the rank of the modified matrix is at most r. [2]

Note that the above algorithm only modifies the entries of $A_{22} \in \mathbb{F}^{(n-r) \times (n-r)}$. Thus, at most $(n-r)^2$ many entries in A are changed, and $\mathcal{R}_A^{\mathbb{F}}(r) \leq (n-r)^2$. □

We will now prove that almost all matrices have rigidity $(n-r)^2$.

Theorem 1.10 (Valiant's lower bounds [Val77])**.** *For any field $\mathbb{F}$,*

- *if $\mathbb{F}$ is infinite, then for all $0 \leq r \leq n$ there exists a matrix $M \in \mathbb{F}^{n \times n}$ of rigidity*

$$\mathcal{R}_M^{\mathbb{F}}(r) = (n-r)^2 \ ;$$

- *if $\mathbb{F}$ is finite, then for all $0 \leq r \leq n - \Omega(\sqrt{n})$ there exists a matrix $M \in \mathbb{F}^{n \times n}$ of rigidity*

$$\mathcal{R}_M^{\mathbb{F}}(r) = \Omega\left((n-r)^2 / \log n\right) \ .$$

Proof of Theorem 1.10. Let $M_{r,s} = \{A \in \mathbb{F}^{n \times n} : \ \mathcal{R}_A^{\mathbb{F}}(r) \leq s\}$ be the set of all matrices of r-rigidity at most s. We will show that the n^2 elements of matrices from $M_{r,s}$ lie in the union of images of a few rational maps from $\mathbb{F}^{n^2+s-(n-r)^2}$ to $\mathbb{F}^{n^2}$. Intuitively, since for $s \ll (n-r)^2$ these images cover only a negligible fraction of all matrices in $\mathbb{F}^{n \times n}$, we will have that "most" of the matrices are rigid.

For every matrix $M \in M_{r,s}$, there exists an s-sparse matrix $S \in \mathbb{F}^{n \times n}$ and a low-rank matrix $L \in \mathbb{F}^{n \times n}, \mathrm{rank}(L) = k \leq r$ such that $M = S + L$. After one of at most $\binom{n}{k}^2$ permutations of rows and columns, we have the first k rows and columns of L linearly independent. The same permutations of rows and columns applied to M, give us a matrix of the form

$$\begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix} \ , \tag{1.11}$$

where $M_{11} \in \mathbb{F}^{k \times k}, M_{12} \in \mathbb{F}^{k \times (n-k)}, M_{21} \in \mathbb{F}^{(n-k) \times k}, M_{22} \in \mathbb{F}^{(n-k) \times (n-k)}$. Moreover, for at least one out of $\binom{n^2}{s}$ choices of s entries of the matrix, we have that a change in those entries makes $\mathrm{rank}(M_{11}) = \mathrm{rank}(M)$. Similarly to Theorem 1.8, this implies that all entries of M_{22} are then rational maps of the entries in

[2] Formally, we set $A_{22} = A_{21} B^{-1} A_{12}$.

M_{11}, M_{12}, M_{21}. That is, the n^2 entries of any matrix $M \in M_{r,s}$ lie in the union of at most $\binom{n}{r}^2 \cdot \binom{n^2}{s}$ rational maps from $\mathbb{F}^{s+n^2-(n-r)^2}$ to $\mathbb{F}^{n^2}$.

When $\mathbb{F}$ is infinite and $s < (n-r)^2$, every matrix in $M_{r,s}$ is in the union of finitely many images of rational functions from $\mathbb{F}^{n^2-1}$ to $\mathbb{F}^{n^2}$. Since n^2 rational functions of n^2-1 variables are algebraically dependent (see, e.g., [For92]), a finite union of such images is the set of roots of a non-zero polynomial. This implies that some matrices from $\mathbb{F}^{n\times n}$ do not belong to $M_{r,s}$.

When $|\mathbb{F}| = q < \infty$ is finite, each $M \in M_{r,s}$ is uniquely specified by one out of $\binom{n}{r}^2$ permutations, one of $\binom{n^2}{s}$ choices of s elements, values of those s elements, and values of the entries in M_{11}, M_{12}, M_{21}. Thus, the size of $M_{r,s}$ is bounded from above by

$$\binom{n}{r}^2 \cdot \binom{n^2}{s} \cdot q^s \cdot q^{n^2-(n-r)^2} \leq 2^{2n+2s\log n} \cdot q^{n^2+s-(n-r)^2},$$

which is at most $o(q^{n^2})$ for every $s < (n-r)^2/\Omega(\log_q n)$ and $r = n - \Omega(\sqrt{n})$. □

Note that the proof of the $\tilde{\Omega}\left((n-r)^2\right)$ lower bound in Theorem 1.10 does not provide a description of a rigid matrix, it merely proves its existence. This brings us to a discussion on explicitness of matrix constructions.

Lecture 3

1.5 On explicitness

In this section we will see two constructions of very rigid matrices. The main drawback of these constructions is that we do not know a polynomial time algorithm outputting the entries of these matrices.

First we show that a matrix consisting of algebraically independent elements has maximal rigidity. (One simple way to construct n^2 algebraically independent elements is given by Lindemann-Weierstrass Theorem.)

Lemma 1.12. *Let $M \in \mathbb{R}^{n\times n}$ be a matrix where all n^2 elements are algebraically independent over $\mathbb{Q}$. Then for every $0 \leq r \leq n$,*

$$\mathcal{R}^{\mathbb{R}}_M(r) = (n-r)^2 .$$

Proof. Let $s = (n-r)^2 - 1$. Assume, for the sake of contradiction, that $\mathcal{R}^{\mathbb{R}}_M(r) \leq s$. Then there exists an s-sparse matrix S, such that $\mathrm{rank}(M+S) \leq r$. Similarly to Theorem 1.10, the n^2 entries of M are rational functions of the s non-zero entries of S and at most $n^2-(n-r)^2$ entries of M. Therefore, polynomials of at most $s + n^2 - (n-r)^2 < n^2$ elements generate all n^2 entries of M, which contradicts the assumption on algebraic independence of the elements of M. This implies that $\mathcal{R}^{\mathbb{R}}_M(r) \geq (n-r)^2$. On the other hand, Theorem 1.8 gives us that $\mathcal{R}^{\mathbb{R}}_M(r) \leq (n-r)^2$. □

While the construction of Lemma 1.12 has optimal rigidity, and each entry of such a matrix may have a very succinct mathematical description, there is no efficient algorithm outputting all digits of these entries. Thus, we will require that *explicit* constructions of matrices have polynomial-time algorithms outputting their entries.

Another non-explicit construction of rigid matrices is via an exponential-time algorithm. Suppose that we have a fixed finite field $\mathbb{F}$ of size $|\mathbb{F}| = q$. Then there is a trivial algorithm which runs in time $q^{O(n^2)}$ and outputs a rigid matrix. Let us fix an $0 \leq r \leq n - \Omega(\sqrt{n})$, and $s = \Omega\left((n-r)^2/\log n\right)$. In time $q^{O(n^2)}$, one can go over all pairs of matrices $M, S \in \mathbb{F}^{n\times n}$. For every such pair, the algorithm checks whether S is s-sparse and $\mathrm{rank}(M+S) \leq r$. When the algorithm finds an M for which there is no S with the above conditions, it outputs M as a rigid matrix and halts. (Theorem 1.10 guarantees existence of such a rigid matrix.)

When the field $\mathbb{F}$ is infinite, an algorithm cannot enumerate all matrices. But even in this case it is possible to construct a rigid matrix in time $2^{O(n^2)}$. In order to prove this, we will first show that there exists a rigid matrix with all entries from $\{0,1\}$. This will give us a way to enumerate all such matrices in time $2^{O(n^2)}$. Next, we will show that given such a matrix, one can check its rigidity in time $2^{O(n^2)}$, which will finish the proof.

Theorem 1.13 ([PR94]). *For all large enough n, there exists a matrix $M \in \{0,1\}^{n\times n}$ such that*

$$\mathcal{R}^{\mathbb{R}}_M\left(\frac{n}{200}\right) \geq \frac{n^2}{100} .$$

Proof. Let $r = \frac{n}{200}$, and assume that all matrices $M \in \{0,1\}^{n\times n}$ have rigidity $\mathcal{R}^{\mathbb{R}}_M(r) \leq s$. We will show the lower bound of $s \geq \frac{n^2}{100}$. Each such matrix M can be written as

$$M = S + L_1 L_2 , \tag{1.14}$$

where $S \in \mathbb{R}^{n\times n}$ is s-sparse, $L_1 \in \mathbb{R}^{n\times r}$ and $L_2 \in \mathbb{R}^{r\times n}$. There are $\binom{n^2}{\leq s}$ ways to choose the set of non-zero entries in S, let us fix one such set Γ. From Equation 1.14, each entry of M is a degree-2 polynomial of the entries of L_1, L_2 and Γ. In particular, there exist a set of n^2 degree-2 polynomials $\{f^{\Gamma}_{ij}\}_{i,j\in[n]}$ with variables being the entries of L_1, L_2, Γ such that $M_{ij} = f^{\Gamma}_{ij}(L_1, L_2, S)$.

For a set of t-variate polynomials $F = \{f_{ij}\}_{i,j\in[n]}$, we define its set of *zero-patterns* as the set of all sequences of zero-non-zero outputs of functions from F:

$$Z(F) = \{M \in \{0,1\}^{n\times n} : \ \exists x \in \mathbb{R}^t \ \forall i,j \in [n], \ M_{i,j} = \mathbf{1}_{f_{ij}(x)\neq 0}\}.$$

We will use the following lemma which asserts that for a set F of low-degree polynomials, $Z(F)$ is small.

Lemma 1.15 ([RBG01]). *If $F = \{f_{ij}\}_{i,j\in[n]}$ is a collection of t-variate polynomials of degree at most d, then*

$$|Z(F)| \leq \binom{t + dn^2}{t} .$$

Proof of Lemma 1.15. Let $m = n^2$ be the number of polynomials, and let $N = |Z(F)|$. Let $x_1, x_2, \ldots, x_N \in \mathbb{R}^t$ be a set of points witnessing the N distinct zero-patterns of F. For an $i \in [N]$, let $S_i \subseteq [m]$ be the set of (indices of) polynomials from F which *are not* zeros at the point x_i. For every $i \in [N]$, define the following polynomial

$$g_i = \prod_{k\in S_i} f_k .$$

Note that $g_i(x_j) = 0$ if and only if there exists $f_k \in S_i \setminus S_j$. Therefore, we have $g_i(x_j) = 0$ if and only if $S_i \not\subseteq S_j$.

Now we prove that all $\{g_i\}_{i\in[N]}$ are linearly independent. Suppose, to the contrary, that there exist $a_1, a_2, \ldots, a_N \in \mathbb{R}$ such that $\sum_{i\in[N]} a_i g_i = 0$, and at least one $a_i \neq 0$. Let

$$i^* = \operatorname*{arg\,min}_{i\in[N],\ a_i\neq 0} |S_i| .$$

We have that $a_{i^*} g_{i^*}(x_{i^*}) \neq 0$ and $\sum_{i\in[N]} a_i g_i(x_{i^*}) = 0$. Due to the minimality of S_{i^*}, for every $a_i \neq 0$, $S_i \not\subseteq S_{i^*}$. This implies that $a_i g_i(x_{i^*}) = 0$ for all $i \neq i^*$, and, thus, $\sum_{i\in[N]} a_i g_i(x_{i^*}) \neq 0$, which leads to a contradiction.

Finally, since the degree of each g_i is at most dm, and all g_i are linearly independent, N is bounded from above by the dimension of the space spanned by t-variate polynomials of degree at most dm. Thus, $N \leq \binom{t+dm}{t}$. □

Recall that from Equation 1.14, all matrices $M \in \{0,1\}^{n\times n}$ can be described by $\{f^{\Gamma}_{ij}\}_{i,j\in[n]}$ for some $\Gamma \in \binom{[n]\times[n]}{s}$, where each polynomial has degree at most 2 and depends on $2rn + s$ variables. Now, from Lemma 1.15 with $t = 2rn + s$ and $d = 2$, we have that

$$\binom{2rn + s + 2n^2}{2rn + s} \cdot \binom{n^2}{\leq s} \geq 2^{n^2} . \tag{1.16}$$

Assume, for the sake of contradiction, that for some $r \leq \frac{n}{200}$ we have $s < \frac{n^2}{100}$. We have that $2rn+s \leq \frac{n^2}{50}$. Now, the left-hand side of Equation 1.16 can be bounded from above as follows:

$$\binom{2rn+s+2n^2}{2rn+s} \cdot \binom{n^2}{\leq s} \leq \binom{\frac{101n^2}{50}}{\frac{n^2}{50}} \cdot \binom{n^2}{\leq \frac{n^2}{100}} \leq (101e)^{\frac{n^2}{50}} \cdot (100e)^{\frac{n^2}{100}} \leq 2^{\frac{n^2}{2}} ,$$

which contradicts Equation 1.16. Thus, we conclude that for any $r \leq \frac{n}{200}$, there exists a matrix $M \in \{0,1\}^{n\times n}$ such that $\mathcal{R}^{\mathbb{R}}_M(r) \geq \frac{n^2}{100}$. □

Now we show that one can check whether a given matrix $M \in \{0,1\}^{n\times n}$ is rigid in time $2^{O(n^2)}$. [3]

Theorem 1.17. *Let $M \in \{0,1\}^{n\times n}$, and r and s be non-negative integers. Then one can decide whether $\mathcal{R}^{\mathbb{R}}_M(r) > s$ in time $2^{O(n^2)}$.*

Proof. Note that $\mathcal{R}^{\mathbb{R}}_M(r) \leq s$ if and only if $M = S + L_1L_2$ for s-sparse S and $L_1 \in \mathbb{R}^{n\times r}, L_2 \in \mathbb{R}^{r\times n}$. For any choice of non-zero entries of S, we have that the entries of M are degree-2 polynomials of $t = 2nr + s$ variables with $\{0,1\}$-coefficients. It is known that deciding whether such a system of polynomial equations has a real solution can be solved in time $2^{O(n^2)}$ (see, e.g., Proposition 13.19 in [BPR07]). Since there are $\binom{n^2}{s} \leq 2^{n^2}$ choices of s non-zero entries, we have that the total running time of the algorithm is $2^{O(n^2)}$. □

This way we have a set of 2^{n^2} matrices such that at least one of them is rigid, and rigidity of each matrix can be checked in time $2^{O(n^2)}$. This gives us a $2^{O(n^2)}$-time algorithm for constructing a rigid matrix over the reals.

Although the above algorithms construct matrices of high rigidity, their running time is $2^{\Omega(n^2)}$. We define *explicit* constructions of matrices as matrices that have algorithms outputting all their entries in *polynomial time*.

1.6 Summary

In Theorem 1.8 we showed that for every field $\mathbb{F}$, matrix $M \in \mathbb{F}^{n\times n}$, and integer $0 \leq r \leq n$, $\mathcal{R}^{\mathbb{F}}_M(r) \leq (n-r)^2$. Below we summarize the non-explicit *lower bounds on rigidity* presented in this chapter.

rigidity	field	running time	reference
$\frac{(n-r)^2}{\log n}$	any finite field	existence	Theorem 1.10
$(n-r)^2$	any infinite field	existence	Theorem 1.10
$(n-r)^2$	$\mathbb{R}$	algebraically independent entries	Lemma 1.12
$\frac{(n-r)^2}{\log n}$	any fixed finite field	$2^{O(n^2)}$	section 1.5
$(n-r)^2$	$\mathbb{R}$	$2^{O(n^2)}$	Theorem 1.13, Theorem 1.17

Table 1.1: Summary of non-explicit lower bounds.

[3] For more efficient algorithms for the case of low rigidity parameters see [FLM+18]. A **PSPACE**-algorithm for this problem follows immediately from the fact that existential theory of the reals lies in **PSPACE** [Can88].

1.7 Notes

Rigidity was introduced as a means to study circuit complexity by Valiant [Val77] and Grigoriev [Gri76]. An excellent presentation of the known lower bounds on rigidity over large fields can be found in the book of Lokam [Lok09]. The books of Jukna [Juk12] and Jukna and Sergeev [JS13] include many applications of matrix rigidity to circuit complexity. Earlier surveys on rigidity are due to Codenotti [Cod00] and Cheraghchi [Che05]. The tight upper bound of Theorem 1.8, and the non-constructive lower bounds of Theorem 1.10 and Theorem 1.13 were proven by Valiant [Val77], and Pudlák and Rödl [PR94]. The proof of Theorem 1.2 was first given by Midrijānis [Mid05].

Chapter 2

Explicit Constructions

In this chapter we give three proofs [Fri93, PR94, SSS97] of the best known explicit lower bound of $\mathcal{R}(r) \geq \Omega\left(\frac{n^2}{r} \cdot \log \frac{n}{r}\right)$ on matrix rigidity. All the three proofs work for (almost) any generator matrix of a good linear code. Since there are explicit linear codes over all finite fields, the presented proofs work for all finite fields. We will see that the last construction (due to Shokrollahi, Spielman and Stemann) easily generalizes to infinite fields. **Lecture 4**

A *linear* code over a field $\mathbb{F}$ is a linear subspace $C \subseteq \mathbb{F}^n$ of dimension k. The *distance* of the code is the minimum Hamming distance between two vectors in C or, equivalently, the minimum Hamming weight of a non-zero vector in C.

Definition 2.1 (Linear code)**.** *Let $\mathbb{F}$ be a field, and n, k, d be positive integers such that $d, k < n$. A subspace $C \subseteq \mathbb{F}^n$ is a linear code of dimension k with minimum distance d if*

1. $\dim(C) = k$;
2. *for every $x \in C \backslash \{0\}$, $\|x\|_0 \geq d$.*

A linear code $C \subseteq \mathbb{F}^n$ of dimension k can be specified by a *generator matrix* $G \in \mathbb{F}^{n \times k}$ such that C is the column space of G. Since we focus on asymptotic behavior of codes, by a code we will mean an infinite sequence $C = \{C_i : i \in \mathbb{N}\}$ where $C_i \subseteq \mathbb{F}^n$. For our purposes, it will suffice to say that a code is *explicit* if there is a polynomial time algorithm that for every n, outputs a generator matrix of C_n in time $\mathsf{poly}(n)$. We will say that a code C is *good* if for the codes of this sequence we have that $k = \Theta(n)$ and $d = \Theta(n)$.

There are exist explicit good linear error correcting codes over all finite fields (see, e.g., Justesen and Goppa codes in [MS77, LG88, vL12]).

Proposition 2.2. *For any finite field $\mathbb{F}$, there exists an explicit family of linear error correcting codes over $\mathbb{F}$ of dimension $k = n/4$ and minimum distance $d = \delta n$ for a constant $\delta > 0$.*

The main result of this chapter is the following.

Theorem 2.3. *Let F be a fixed finite field, and $G \in \mathbb{F}^{n \times k}$ be a generator matrix of a linear code of dimension $k = \Theta(n)$ and distance $d = \Theta(n)$, then for every $\Omega(\log n) < r < O(n)$,*

$$\mathcal{R}_G^{\mathbb{F}}(r) \geq \Omega\left(\frac{n^2}{r} \cdot \log \frac{n}{r}\right) .$$

2.1 The lower bound of Friedman

Let us fix a generator matrix $G \in \mathbb{F}^{n \times k}$ of a good linear code with distance and dimension $d, k = \Theta(n)$. We will prove the lower bound of Friedman in two steps. First, in Theorem 2.4 we will show that G has high "column rigidity". That is, in order to drop the rank of G to r, one has to modify at least $\Omega(\frac{n}{r} \log_q \frac{k}{r})$ entries in *some* column of G. Second, in Theorem 2.5 we will use a simple averaging argument to reduce column rigidity to rigidity.

Theorem 2.4 ([Fri93]). *Let $\mathbb{F} = \mathbb{F}_q$ be a finite field of size q. Let $G \in \mathbb{F}^{n\times k}$ be a generator matrix of a code of dimension k and distance δn for a constant $0 < \delta < 1$. For any $\log_q k \le r \le \frac{k}{4}$, if every column of $B \in \mathbb{F}^{n\times k}$ contains at most $\frac{\delta n}{4r}\log_q \frac{k}{r}$ non-zero entries, then*

$$\operatorname{rank}(G+B) > r \ .$$

Proof of Theorem 2.4. Assume, for the sake of contradiction, that there exists $B \in \mathbb{F}_q^{n\times k}$ such that $\operatorname{rank}(G+B) \le r$ and each column of B has at most $\frac{\delta n}{4r}\log_q \frac{k}{r}$ nonzero entries.

The proof employs two ideas. First, using a packing argument, we will show that the *kernel* of $G+B$ must contain a sparse vector $x \in \mathbb{F}^k$. Second, as Gx is a codeword of C and $Gx + Bx = \mathbf{0}$, Bx is also a codeword of C and, thus, $\|Bx\|_0$ must be large due to the minimum distance property of C. This leads to a contradiction as x and the columns of B are sparse.

Let us draw a Hamming ball of radius $d/2$ around each point in the kernel of $G+B$. Since we assume that $\operatorname{rank}(\ker(G+B)) = k - \operatorname{rank}(G+B) \ge k - r$, if

$$q^{k-r} \cdot \left|\text{Hamming ball of radius } d/2 \text{ in } \mathbb{F}_q^k\right| > q^k,$$

then there must be two distinct points in the null space of $G+B$ such that their Hamming balls intersect. This gives us a non-zero vector x in the kernel of $G+B$ of sparsity at most d. The following shows that it suffice to pick d as an even number between $\frac{2r}{\log_q \frac{k}{r}} \le d \le \frac{2r}{\log_q \frac{k}{r}} + 2$.

$$\begin{aligned}\left|\text{Hamming ball of radius } d/2 \text{ in } \mathbb{F}_q^k\right| &\ge \binom{k}{d/2}\cdot(q-1)^{d/2} \\ &\ge q^{\frac{d}{2}\cdot\log_q\left[\frac{2k}{d}(q-1)\right]} > q^r \ .\end{aligned}$$

Next, since Gx is a *non-zero* codeword of C and $Gx + Bx = \mathbf{0}$, we know that Bx is a non-zero codeword of C and, thus, $\|Bx\|_0 \ge \delta n$. On the other hand, x has only d non-zero coordinates, and each column of B has at most $\frac{\delta n}{4r}\log_q \frac{k}{r}$ non-zero entries. We have that

$$\|Bx\|_0 \le d \cdot \frac{\delta n}{4r}\log_q \frac{k}{r} \le \left(\frac{2r}{\log_q \frac{k}{r}} + 2\right)\frac{\delta n}{4r}\log_q \frac{k}{r} < \delta n \ ,$$

which contradicts the distance property of C. □

Theorem 2.5. *Let $\mathbb{F} = \mathbb{F}_q$ be a finite field of size q. Let $G \in \mathbb{F}^{n\times k}$ be a generator matrix of a code of dimension k and distance δn for a constant $0 < \delta < 1$. Then for any $\frac{\log_q k}{2} \le r \le \frac{k}{8}$,*

$$\mathcal{R}_G^{\mathbb{F}}(r) \ge \frac{\delta k n \log_q \frac{k}{2r}}{8r} \ .$$

Proof. Assume, for the sake of contradiction, that there exists $S \in \mathbb{F}_q^{n\times k}$ such that $\operatorname{rank}(G+S) \le r$ and $\|S\|_0 \le \frac{\delta k n \log_q \frac{k}{2r}}{8r}$. Let $J \subset [k]$ be the indices of the $\frac{k}{2}$ sparsest columns of S, and let S_J be the sub-matrix of S restricted to the columns in J. By Markov's inequality, each column of S_J has at most

$$\left(\frac{\delta k n \log_q \frac{k}{2r}}{8r}\right) / \left(\frac{k}{2}\right) = \frac{\delta n}{4r}\log_q \frac{k}{2r}$$

many non-zero entries. Now Theorem 2.4 applied to G_J and S_J implies that a column in S_J must contain more than $\frac{\delta n}{4r}\log_q \frac{k}{2r}$ non-zero entries, which leads to a contradiction. □

Now, using good codes from Proposition 2.2, we get a lower bound of $\mathcal{R}_G^{\mathbb{F}}(r) \ge \Omega\left(\frac{n^2}{r}\cdot\log\frac{n}{r}\right)$ for any $\Omega(\log_q n) \le r \le \frac{n}{32}$.

Bibliography

[Blu83] Norbert Blum. A Boolean function requiring $3n$ network size. *Theoretical Computer Science*, 28(3):337–345, 1983.

[Bop97] Ravi B. Boppana. The average sensitivity of bounded-depth circuits. *Inf. Process. Lett.*, 63(5):257–261, 1997.

[BPR07] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in real algebraic geometry*, volume 10. Springer Science & Business Media, 2007.

[Can88] John Canny. Some algebraic and geometric computations in pspace. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC 1988)*, pages 460–467. ACM, 1988.

[Cha94a] Aleksandr V. Chashkin. On the complexity of Boolean matrices, graphs and their corresponding Boolean functions. *Diskretnaya matematika*, 4(3):229–257, 1994.

[Cha94b] Aleksandr V. Chashkin. On the complexity of Boolean matrices, graphs and their corresponding Boolean functions. *Discrete Math. and Appl.*, 4(3):229–257, 1994.

[Che05] Mahdi Cheraghchi. On matrix rigidity and the complexity of linear forms. *Electronic Colloquium on Computational Complexity (ECCC)*, 2005.

[Cod00] Bruno Codenotti. Matrix rigidity. *Linear Algebra and its Applications*, 304(1-3):181–192, 2000.

[DK11] Evgeny Demenkov and Alexander S Kulikov. An elementary proof of a $3n - o(n)$ lower bound on the circuit complexity of affine dispersers. In *Proceedings of the 36th Internationl Symposium on the Mathematical Foundations of Computer Science (MFCS 2011)*, pages 256–265. Springer, 2011.

[EGS76] Paul Erdös, Ronald L. Graham, and Endre Szemerédi. On sparse graphs with dense long paths. *Computers and Mathematics with Applications*, pages 365–369, 1976.

[FGHK16] Magnus Find, Alexander Golovnev, Edward A. Hirsch, and Alexander S. Kulikov. A better-than-$3n$ lower bound for the circuit complexity of an explicit function. In *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2016)*, pages 89–98. IEEE, 2016.

[FLM+18] Fedor V. Fomin, Daniel Lokshtanov, Syed Mohammad Meesum, Saket Saurabh, and Meirav Zehavi. Matrix rigidity from the viewpoint of parameterized complexity. *SIAM Journal on Discrete Mathematics*, 32(2):966–985, 2018.

[For92] Krister Forsman. Two themes in commutative algebra: Algebraic dependence and kähler differentials, 1992.

[Fri93] Joel Friedman. A note on matrix rigidity. *Combinatorica*, 13(2):235–239, 1993.

[Gri76] Dmitrii Yu. Grigoriev. Application of separability and independence notions for proving lower bounds of circuit complexity. *Zapiski Nauchnykh Seminarov POMI*, 60:38–48, 1976.

[Hås86] Johan Håstad. Almost optimal lower bounds for small depth circuits. In *STOC 1986*, pages 6–20, 1986.

[HJP93] Johan Håstad, Stasys Jukna, and Pavel Pudlák. Top-down lower bounds for depth 3 circuits. In *FOCS 1993*, pages 124–129, 1993.

[IM02] Kazuo Iwama and Hiroki Morizumi. An explicit lower bound of $5n - o(n)$ for Boolean circuits. In *Proceedings of the 27th Internationl Symposium on the Mathematical Foundations of Computer Science (MFCS 2002)*, pages 353–364. Springer, 2002.

[JS13] Stasys Jukna and Igor Sergeev. Complexity of linear boolean operators. *Foundations and Trends in Theoretical Computer Science*, 9(1):1–123, 2013.

[Juk12] Stasys Jukna. *Boolean function complexity: advances and frontiers*, volume 27. Springer Science & Business Media, 2012.

[LG88] Jacobus Hendricus van Lint and Gerard van der Geer. *Introduction to coding theory and algebraic geometry*. Birkhäuser Basel, 1988.

[Lok09] Satyanarayana V. Lokam. Complexity lower bounds using linear algebra. *Foundations and Trends in Theoretical Computer Science*, 4(1–2):1–155, 2009.

[LR01] Oded Lachish and Ran Raz. Explicit lower bound of $4.5n-o(n)$ for Boolean circuits. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC 2001)*, pages 399–408. ACM, 2001.

[Lup56] Oleg B. Lupanov. On rectifier and switching-and-rectifier schemes. In *Dokl. Akad. Nauk SSSR*, volume 111, pages 1171–1174, 1956. In Russian.

[Lup59a] Oleg B. Lupanov. A method of circuit synthesis. *Izv. VUZov, Radiofizika*, 1:120–140, 1959. In Russian.

[Lup59b] Oleg B. Lupanov. A method of circuit synthesis. *Izv. VUZov, Radiofizika*, 1:120–140, 1959. In Russian.

[Mid05] Gatis Midrijānis. Three lines proof of the lower bound for the matrix rigidity. *arXiv preprint cs/0506081*, 2005.

[MS77] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error-correcting codes*. Elsevier, 1977.

[MW17] Or Meir and Avi Wigderson. Prediction from partial information and hindsight, with application to circuit lower bounds. In *ECCC*, volume 24, 2017.

[Nec66] Edward I. Nechiporuk. On a Boolean function. *Dokl. Akad. Nauk SSSR*, 169(4):765–766, 1966.

[PPSZ05] Ramamohan Paturi, Pavel Pudlák, Michael E Saks, and Francis Zane. An improved exponential-time algorithm for k-SAT. *J. ACM*, 52(3):337–364, 2005.

[PPZ97] Ramamohan Paturi, Pavel Pudlák, and Francis Zane. Satisfiability coding lemma. In *FOCS 1997*, pages 566–574, 1997.

[PR94] Pavel Pudlák and Vojtech Rödl. Some combinatorial-algebraic problems from complexity theory. *Discrete Mathematics*, 1(136):253–279, 1994.

[RBG01] Lajos Rónyai, László Babai, and Murali Ganapathy. On the number of zero-patterns of a sequence of polynomials. *Journal of the American Mathematical Society*, 14(3):717–735, 2001.

[Sha49a] Claude E. Shannon. The synthesis of two-terminal switching circuits. *Bell Syst. Tech. J.*, 28:59–98, 1949.

[Sha49b] Claude E. Shannon. The synthesis of two-terminal switching circuits. *Bell Syst. Tech. J.*, 28:59–98, 1949.

[SSS97] Mohammad A. Shokrollahi, Daniel A. Spielman, and Volker Stemann. A remark on matrix rigidity. *Information Processing Letters*, 64(6):283–285, 1997.

[Val77] Leslie G. Valiant. Graph-theoretic arguments in low-level complexity. In *Proceedings of the 2nd Internationl Symposium on the Mathematical Foundations of Computer Science (MFCS 1977)*, pages 121–127. Springer, 1977.

[Vio09] Emanuele Viola. On the power of small-depth computation. *Foundations and Trends in Theoretical Computer Science*, 5(1):1–72, 2009.

[vL12] Jacobus Hendricus van Lint. *Introduction to coding theory*, volume 86. Springer Science & Business Media, 2012.