CSC 2429 – Approaches to the P versus NP Question

Lecture #9: 19 March 2014

Lecturer: Mika Göös

Scribe Notes by: Mika Göös

This lecture consist of two parts:

(1) We introduce Valiant's [Val77] concept of *matrix rigidity* and prove a theorem of Valiant that relates rigidity to proving size lower bounds for log-depth circuits.

(2) We discuss a theorem of Razborov [Raz89] (following [Wun12]) that interprets rigidity as a complexity measure in the field of communication complexity. In particular, the resulting measure is stronger than the analogue of the polynomial hierarchy in communication complexity (for which no explicit lower bounds are known).

1 Matrix Rigidity

Fix some field $\mathbb{F}$ (e.g., $\mathbb{F} = \mathrm{GF}(2)$, for simplicity). Informally, a matrix $A \in \mathbb{F}^{n\times n}$ is *rigid* if it is far from all low-rank matrices in Hamming distance. More formally, letting $|B|$ denote the number of non-zero elements in a matrix $B \in \mathbb{F}^{n\times n}$ the rigidity of A is defined by

$$\mathrm{Rig}_A(r) = \min\{|B| : \mathrm{rank}(A+B) \leq r\}.$$

In words, $\mathrm{Rig}_A(r) = k$ iff k is the minimum number of entries of A that need to be modified in order to bring the rank of A down to r.

Warm-up. It is an easy exercise to prove that for all A we have $\mathrm{Rig}_A(r) \leq (n-r)^2$. This is nearly tight by a direct counting argument: Valiant [Val77, Theorem 6.4] proves that $\mathrm{Rig}_A(r)$ is at least about $(n-r)^2/\log n$ for a random A (and all fields $\mathbb{F}$). This shows that rigid matrices exist in great numbers—however, as always, the challenge is to exhibit an *explicit* such matrix.

Currently, the best explicit construction over finite fields is due to Friedman [Fri93] (see also Jukna [Juk12, §13.8]) that achieves rigidity of about $\geq n^2/r\log(n/r)$. Here we restrict to giving a simpler construction[1] due to Midrijānis [Mid05] that comes close to this: Suppose n is divisible by $2r$ and construct A as the block matrix having $n^2/(2r)^2$ blocks, each containing a copy of the $2r \times 2r$ identity matrix. Then $\mathrm{Rig}_A(r) \geq n^2/4r$ since in order to reduce the rank of A to at most r we need to reduce the rank of each $2r \times 2r$ identity block to at most r and this requires modifying at least r entries in each block.

[1] See also some related discussion in the *Computational Complexity* blog at: http://blog.computationalcomplexity.org/2005/07/matrix-rigidity.html.

2 Lower bounds against log-depth circuits

Next, we study *linear circuits* that compute a linear map $\boldsymbol{x} \mapsto A\boldsymbol{x}$ defined by some $A \in \mathbb{F}^{n\times n}$. Here $\boldsymbol{x} = (x_1, \ldots, x_n)$ is a vector in $\mathbb{F}^n$ and the inputs to the linear circuit are the n variables x_i, $i \in [n]$. The task is to compute the n outputs of the vector $A\boldsymbol{x}$ by using only *linear gates*, namely +-gates of fan-in 2 and also gates that perform multiplications by constants in $\mathbb{F}$.

Which matrices A are hard to compute in this setting? We prove the following result of Valiant stating that rigid matrices are hard for log-depth circuits.

Theorem 1 (Valiant [Val77])**.** *Suppose that $A \in \mathbb{F}^{n\times n}$ can be computed with a linear circuit of size $O(n)$ and depth $O(\log n)$. Then $\mathrm{Rig}_A(O(n/\log\log n)) \leq n^{1+\epsilon}$ for any constant $\epsilon > 0$.*

Indeed, contrapositively, if we could exhibit a matrix A such that $\mathrm{Rig}_A(O(n/\log\log n)) \geq n^{1+\epsilon}$ for some constant $\epsilon > 0$ then Theorem 1 implies that A cannot be computed in simultaneous size $O(n)$ and depth $O(\log n)$. This would solve a long-standing open problem. Note in particular that we currently lack explicit matrices with these required rigidity properties.

We need the following lemma in the proof.

Lemma 2. *Given a DAG G with m edges and depth d there is a set of $m/\log d$ edges such that by deleting them we obtain a graph of depth at most $d/2$.*

Proof. A *proper labelling* of G is a labelling of the nodes with nonnegative numbers such that if (u, v) is an edge, then the label of u is smaller than the label of v. Since G has depth d there exists a *canonical* proper labelling using labels from $[0, d]$. Partition the edges of G into $\log d$ colour classes as follows: an edge (u, v) has colour $i \in [0, \log d]$ if the most significant bit where the labels of u and v differ (when written in binary) is the i-th from the left. Delete the edges in the color class that has the fewest edges; say this is the i-th class. We claim that the resulting graph has depth at most $d/2$. Indeed, consider the node labelling obtained from the canonical one by deleting the i-th bit. The resulting labelling uses labels from $[0, d/2]$ and it is easy to check that this labelling is proper, which proves the claim. □

Proof of Theorem 1. Let C be a linear circuit for A of size $O(n)$ and depth $O(\log n)$. By applying Lemma 2 some $O(1)$ many times we can find a set of wires E, $|E| = O(n/\log\log n)$, such that their removal from C decreases the depth below $\epsilon \log n$. To prove the theorem it suffices to express A as

$$A = B + H.$$

where B is sparse, $|B| \leq n^{1+\epsilon}$, and H has low rank, $\mathrm{rank}(H) = O(n/\log\log n)$.

Low-rank matrix H: For each wire $e \in E$ we associate a rank-1 matrix H_e defined as follows. The value computed by e on input $\boldsymbol{x} = (x_1, \ldots, x_n)$ is some linear combination $\langle \boldsymbol{\alpha}_e, \boldsymbol{x}\rangle$ of the input variables where $\boldsymbol{\alpha}_e \in \mathbb{F}^n$. We define the j-th row of H_e as $\beta_j \boldsymbol{\alpha}_e$ where $\beta_j \in \mathbb{F}$ is the coefficient with which the value of e appears in the j-th output node. (More precisely, $\beta_j = \sum_p \beta_{j,p}$ where the sum is taken over all paths p from e to the j-th output and $\beta_{j,p}$ is the product of the constants appearing on the multiplication gates in p.) Set $H = \sum_e H_e$ so that $\mathrm{rank}(H) = O(n/\log\log n)$.

Sparse matrix B: We define B similarly to H except now we look at the input variables instead of the edges in E. Namely, we define B by letting its j-th row be equal to the vector of coefficients with which the input variables appear in the j-th output in the reduced circuit $C \setminus E$ (where we now think of the edges E as just outputting the value 0). Because the depth

of $C \setminus E$ is only $\epsilon \log n$ each row of B will have at most n^ϵ many non-zero entries. This yields $|B| \leq n^{1+\epsilon}$ as required. □

3 Rigidity and communication complexity

Next we highlight a connection due to [Raz89, Wun12] that relates rigidity and an analogue of the polynomial hierarchy in communication complexity. In this section we assume some familiarity with basic two-party communication complexity; see, e.g., the textbook [KN97].

One of the most fundamental complexity classes in communication complexity is NP. This is the the class of boolean $2^n \times 2^n$ matrices A (the columns and rows of A are indexed by $\{0,1\}^n$) whose 1-entries can be covered using $2^{\mathrm{polylog}(n)}$ many rectangles. (We just abused language: by a matrix $A \in \mathsf{NP}$ we actually mean a *sequence* of matrices, one for each n.) We also define $\oplus\mathsf{P}$ as the class of matrices whose rank over GF(2) is at most $2^{\mathrm{polylog}(n)}$. Equivalently, $A \in \oplus\mathsf{P}$ iff there is a set of $2^{\mathrm{polylog}(n)}$ many rectangles R_i such that every 1-entry of A is covered by *odd* many of the R_i and each 0-entry of A is covered by *even* many of the R_i.

We can use *class operators* to define new classes out of existing ones in perfect analogy to classical (i.e., poly-time Turing machine) complexity theory. Let $\mathcal{C}$ be any class of matrices.

- *Complements:* $A \in \mathrm{co} \cdot \mathcal{C}$ iff the boolean complement of A is in $\mathcal{C}$.
- *Nondeterminism:* $A \in \exists \cdot \mathcal{C}$ iff there are $2^{\mathrm{polylog}(n)}$ many matrices $B_i \in \mathcal{C}$ such that

$$A = \bigvee_i B_i \quad \text{(entry-wise)}.$$

  (Strictly speaking, here we are implicitly requiring that the complexities of the B_i are upper bounded *uniformly*, i.e., they have have the same complexity, e.g., $\leq 2^{\log^C(n)}$.)
- *Conondeterminism:* $\forall \cdot \mathcal{C} = \mathrm{co} \cdot \exists \cdot \mathrm{co} \cdot \mathcal{C}$.
- The *polynomial hierarchy* PH is defined as the union

$$\mathsf{PH} = \bigcup_{k \geq 1} \Sigma_k,$$

  where $\Sigma_1 = \mathsf{NP}$ and $\Sigma_{k+1} = \exists \cdot \mathrm{co} \cdot \Sigma_k$.
- *Bounded-error probability operator:* $A \in \mathrm{BP} \cdot \mathcal{C}$ iff there is a probability distribution μ on matrices $B \in \mathcal{C}$ (whose complexities are uniformly bounded) such that for all $x, y \in \{0,1\}^n$

$$\Pr_{B \sim \mu}[A(x,y) = B(x,y)] \geq 2/3.$$

Proving lower bounds against NP is easy: any covering of the 1-entries of the $2^n \times 2^n$ identity matrix I requires 2^n rectangles so that $I \notin \mathsf{NP}$. By contrast, no explicit matrices are known that are outside Σ_2! It is a long-standing open problem to prove lower bounds against PH or even Σ_2. Razborov's result states that such lower bounds would follow from sufficiently rigid matrices.

Theorem 3 ([Raz89, Wun12])**.** *Suppose* $A \in \mathsf{PH}$. *Then (over* GF(2)*)*

$$\mathrm{Rig}_A(2^{\mathrm{polylog}(n)}) \leq 2^{2n - \mathrm{polylog}(n)}.$$

We do not give a full proof of Theorem 3 here, but only sketch the main ideas following Wunderlich [Wun12]. (See also [Juk12, Theorem 12.39] for an English translation of Razborov's orignal proof.)

First, in classical complexity theory, (the first part of) Toda's theorem [Tod91] states that

$$\mathsf{PH} \subseteq \mathrm{BP} \cdot \oplus\mathsf{P}. \tag{1}$$

It turns out that Toda's proof translates rather directly to the communication complexity setting so that (1) continues to hold for the classes as defined above. In fact, since two-party communication is a *non-uniform* model of computation, many of the ideas in Toda's proof can be simplified as efficiency is not a concern.

Second, by definition, each matrix $A \in \mathrm{BP} \cdot \oplus\mathsf{P}$ can be approximately represented as a probability distribution over rank-$2^{\mathrm{polylog}(n)}$ matrices. By averaging, there must exist a setting of randomness yielding a fixed matrix $B \in \oplus\mathsf{P}$ that agrees with A on at least a $2/3$ fraction of entries.[2] This implies that A can be written as

$$A = B + C, \quad \text{where } |C| \leq 1/3 \cdot 2^{2n},$$

which gives

$$\mathrm{Rig}_A(2^{\mathrm{polylog}(n)}) \leq 1/3 \cdot 2^{2n}.$$

If we had first *amplified* the success probability of the $\mathrm{BP} \cdot \oplus\mathsf{P}$ matrix to $1 - 2^{-\mathrm{polylog}(n)}$ (which is doable inside $\mathrm{BP} \cdot \oplus\mathsf{P}$), we would obtain the bound claimed in Theorem 3.

References

[Fri93] Joel Friedman. A note on matrix rigidity. *Combinatorica*, 13(2):235–239, 1993. doi:10.1007/BF01303207.

[Juk12] Stasys Jukna. *Boolean Function Complexity: Advances and Frontiers*, volume 27 of *Algorithms and Combinatorics*. Springer, 2012.

[KN97] Eyal Kushilevitz and Noam Nisan. *Communication Complexity*. Cambridge University Press, 1997.

[Mid05] Gatis Midrijānis. Three lines proof of the lower bound for the matrix rigidity, 2005. URL: http://arxiv.org/abs/cs/0506081.

[Raz89] Alexander Razborov. On rigid matrices (in Russian). Unpublished manuscript, 1989.

[Tod91] Seinosuke Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991. doi:10.1137/0220053.

[Val77] Leslie G. Valiant. Graph-theoretic arguments in low-level complexity. In Jozef Gruska, editor, *Mathematical Foundations of Computer Science*, volume 53 of *Lecture Notes in Computer Science*, pages 162–176. Springer, 1977. doi:10.1007/3-540-08353-7_135.

[Wun12] Henning Wunderlich. On a theorem of Razborov. *Computational Complexity*, 21(3):431–477, 2012. doi:10.1007/s00037-011-0021-5.

[2] Here we just gave the argument for *Yao's principle* that relates the average-case complexity of a problem with its randomised complexity; see http://en.wikipedia.org/wiki/Yao's_principle for more details.