Barrington’s Theorem

Norbert Lis, based on lecture by Prof. Eric Allender

March 9-11, 1998

1 Barrington’s Theorem

Theorem: $NC^1 = WIDTH(O(1))_SIZE(n^{O(1)})$ branching programs.

Proof.
1) $\supseteq$

Let $A$ be accepted by a $WIDTH(k)$ branching program $BP$ of size $n^k$. View the branching program $BP$ as a sequence of pairs of functions $(f_{i,0}, f_{i,1}), \ldots, (f_{n^k,0}, f_{n^k,1})$, where $f_{i,b} : [1..k] \rightarrow [1..k]$, and for $i$-th symbol $b \in \{0,1\}$ of input $x$, $f_{i,b}$ is picked. Denote selected $f_{i,b}$ to be just $f_i$. Then the BP running on input $x$ can be expressed as function $f = f_1 \circ f_2 \circ \ldots \circ f_{n^k} = \prod_{i=1}^{n^k} f_i$. We want to find an algorithm which would answer the question: is $f(1) = acc$? Here it is:

On input $x$:

$\exists$ guess $f = \prod_{i=1}^{n^k} f_i$ as a $k \times k$ matrix (this can be done in constant time)

call $VERIFY(f, 1, n)$

end

$VERIFY(g, i, j)$:

if $i + 1 = j$ then

return true iff position $i$ in the branching program evaluates to $g$

else

$\exists$ guess $f_a, f_b$ such that $f_a \circ f_b = g$

$\forall$ call $VERIFY(f_a, i, \frac{i+j}{2})$ and $VERIFY(f_b, \frac{i+j}{2} + 1, j)$

end

2) $\subseteq$

Let $A \in NC^1$. We will build a $WIDTH(5)$ branching program for $A$. This branching program will be a permutation program in the sense that each $f_{i,0}, f_{i,1}$ will be a permutation on $[1..5]$. It will have the property: "$\exists$ a 5-cycle $\delta$, such that $x \in A \iff \prod_{i=1}^{n^k} f_i = \delta$, and $x \notin A \iff \prod_{i=1}^{n^k} f_i = i$". This defines what it means for a branching program to $\delta$-recognize $A$. To proceede with the proof, first note the following:
1. If \( \exists \) a BP that \( \delta \)-recognizes \( A \), then \( \exists \) a BP that \( \delta' \)-recognizes \( A \), for any 5-cycle \( \delta' \). This can be justified by noting that \( \exists \) a 5-permutation \( \theta \) such that \( \delta' = \theta \delta \theta^{-1} \) (because any two 5-cycles are isomorphic) and replacing each \( f_{i,b} \) in original BP with \( \theta f_{i,b} \theta^{-1} \).

2. If \( A \) can be \( \delta \)-recognized by a \( \mathit{SIZE} (s(n)) \mathit{WIDTH} (5) \) permutation BP, then so can the complement \( \overline{A} \). Building such BP for \( \overline{A} \) can be accomplished by replacing \( f_{s(n),b} \) with \( \delta^{-1} \circ f_{s(n),b} \) in the original machine. This results in a machine which \( \delta^{-1} \)-recognizes \( \overline{A} \).

3. There exist 5-cycles \( \delta, \pi, \rho \) such that \( \rho = \delta \pi \delta^{-1} \pi^{-1} \), namely, \( \delta = (1, 2, 3, 4, 5) \), \( \pi = (1, 3, 5, 4, 2) \), \( \rho = (1, 3, 2, 5, 4) \).

To complete the proof the following statement will be proved by induction on \( d \): “If \( A \) has \( \mathit{DEPTH} (d) \mathit{NC}^1 \) circuits, then \( A \) is \( \rho \)-recognized by a \( \mathit{WIDTH} (5) \mathit{SIZE} (4^d) \) BP”.

**Basis:** If \( d = 0 \), then one of the input gates is also an output gate, call that gate \( G \). If \( G = x_i \), then let BP be \( f_{i,1} = \rho, f_{i,0} = i \). If \( G = \overline{f_{j,0}} \), then let BP be \( f_{j,0} = \rho, f_{j,1} = i \).

**Induction:** Assume that all \( \mathit{NC}^1 \) circuits with depth \( d' < d \) have corresponding \( \rho \)-BP’s of \( \mathit{WIDTH} (5) \mathit{SIZE} (4^{d'}) \). Further, assume the output gate of circuit \( C_n \) of depth \( d \) for \( A \) is an \( \wedge \)-gate, call it \( G \). Let the language recognized by the sub-circuit attached to the left in-edge of \( G \) be \( A_L \) and the language recognized by the sub-circuit attached to the right in-edge of \( G \) be \( A_R \). By the induction hypothesis, let \( P_\delta \) be a \( \mathit{SIZE} (4^{d-1}) \) BP that \( \delta \)-recognizes \( A_L \), \( P_\pi \) be a \( \mathit{SIZE} (4^{d-1}) \) BP that \( \pi \)-recognizes \( A_R \), \( P_\delta \circ = \mathit{SIZE} (4^{d-1}) \) BP that \( \delta^{-1} \)-recognizes \( A_L \), \( P_\pi \circ = \mathit{SIZE} (4^{d-1}) \) BP that \( \pi^{-1} \)-recognizes \( A_R \).

Since \( A = A_L \cap A_R \), \( P = P_\delta P_\pi P_\delta \circ P_\pi \circ \) recognizes \( A \) and has size \( 4^d \). Since \( \rho = \delta \pi \delta^{-1} \pi^{-1} \), \( P \) \( \rho \)-recognizes \( A \). The case when the output gate is a \( \neg \)-gate is trivial by fact 2 above. Similarly, the case when the output gate is an \( \vee \)-gate reduces to the first two cases by DeMorgan’s Law: \( (p \lor q) = \overline{p} \land \overline{q} \).

2. Completeness

**Definition:** Let \( C \) be a class of functions, and \( A, B \) be languages. We say \( A \) is many-one \( C \)-reducible to \( B \) (denoted \( A \leq^C_m B \)) if \( \exists f \in C \forall x \in A \iff f(x) \in B \).

**Definition:** Let \( D \) be a class of languages, and \( A \) be a language. We say \( A \) is hard for \( D \) under \( \leq^D_m \) if \( \forall B \in D \, B \leq^D_m A \).

**Definition:** Let \( D \) be a class of languages, and \( A \) be a language. \( A \) is complete for \( D \) under \( \leq^D_m \) if \( A \) is hard and \( A \in D \).

**Notes:**

1. Important reducibilities: \( \leq^P_m, \leq^{log}_m, \leq^{AC^0}_m \).
2. Notion of hardness is useful for proving lower bounds. Using diagonalization or some other technique, a set $B$ in some class $\mathcal{D}$ is defined, such that $B$ is very complex. (Usually, $B$ will look very artificial and intrinsically uninteresting.) However, the class $\mathcal{D}$ will usually have some natural and interesting complete sets. Since $B$ is complex, all of the complete sets will also be complex.

3. Many natural problems are complete for some well known complexity class under $\leq^{AC^0}_m$.

**Corollary:** There exists a regular set that is complete for $NC^1$ under $\leq^{AC^0}_m$.

**Proof.**

Let $W_5 = \{\pi_1, ..., \pi_n \mid \pi_1 \circ ... \circ \pi_n = i, \text{ and each } \pi_i \text{ is a permutation on } [1..5]\}$. Clearly, $W_5$ is regular. The regular set that is complete for $NC^1$ under $\leq^{AC^0}_m$ is $\overline{W_5}$. Let $B \in NC^1$. Then there is a dlogtime-uniform $NC^1$ circuit family $C_n$, and on input $x$, let $\pi_i$ be the $i$'th instruction in the branching program for $C_{|x|}$. Then $x$ is accepted by $C_{|x|}$ if and only if $\pi_1, ..., \pi_n \in \overline{W_5}$. 