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^l. View the branching program BP as a sequence of pairs of functions $(f_{1,0}, f_{1,1}), ..., (f_{n^{l'},0}, f_{n^{l'},1})$, where $f_{i,b} : [1..k] \to [1..k]$, 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 $f = f_1 \circ f_2 \circ ... \circ f_{n^{l'}} = \prod_{i=1}^{n^{l'}} f_i$. We want to find an algorithm which would answer the question: is $f(1) = acc$? Here it is:

On input x:
  $\exists$ guess $f = \prod_{i=1}^{n^{l'}} f_i$ 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 $VERIFY(f_b, \frac{i+1}{2} + 1, j)$
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 δ, such that $x \in A \iff \prod_{i=1}^{n^{l'}} f_i = \delta$, and $x \notin A \iff \prod_{i=1}^{n^{l'}} f_i = i$". This defines what it means for a branching program to δ-recognize A. To proceede with the proof, first note the following:

1. If $\exists$ a BP that δ-recognizes A, then $\exists$ a BP that δ'-recognizes A, for any 5-cycle δ'. This can be justified by noting that $\exists$ a 5-permutation θ 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 δ-recognized by a $SIZE(s(n))WIDTH(5)$ permutation BP, then so can the complement $\overline{A}$. Building such BP for $\overline{A}$ can be acomplished by replacing $f_{s(n),b}$ with $\delta^{-1} \circ f_{s(n),b}$ in the original machine. This results in a machine which δ^{-1}-recognizes $\overline{A}$.

3. There exist 5-cycles δ, π, ρ 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 $DEPTH(d)$ NC^1 circuits, then A is ρ-recognized by a $WIDTH(5)SIZE(4^d)$ 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 $f_{1,1} = \rho, f_{1,0} = i$. If $G = \overline{x_j}$, then let BP be $f_{1,0} = \rho, f_{1,1} = i$.

INDUCTION: Assume that all NC^1 circuits with depth $d' < d$ have corresponding ρ-BP's of $WIDTH(5)SIZE(4^{d'})$. 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_δ be a $SIZE(4^{d-1})$ BP that δ-recognizes A_L,
P_π be a $SIZE(4^{d-1})$ BP that π-recognizes A_R,
$P_{\delta^{-1}}$ be a $SIZE(4^{d-1})$ BP that δ^{-1}-recognizes A_L,
$P_{\pi^{-1}}$ be a $SIZE(4^{d-1})$ BP that π^{-1}-recognizes A_R.
Since $A = A_L \cap A_R$, $P = P_\delta P_\pi P_{\delta^{-1}} P_{\pi^{-1}}$ recognizes A and has size 4^d. Since $\rho = \delta\pi\delta^{-1}\pi^{-1}$, P ρ-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: $(p \vee q) = \overline{\overline{p} \wedge \overline{q}}$.

2 Completeness

Definition: Let $\mathcal{C}$ be a class of functions, and A, B be languages. We say A is many-one $\mathcal{C}$-reducible to B (denoted $A \leq_m^{\mathcal{C}} B$) if $\exists_{f \in \mathcal{C}} \forall_x \; x \in A \iff f(x) \in B$.

Definition: Let $\mathcal{D}$ be a class of languages, and A be a language. We say A is hard for $\mathcal{D}$ under $\leq_m^{\mathcal{D}}$ if $\forall_{B \in \mathcal{D}} \; B \leq_m^{\mathcal{D}} A$.

Definition: Let $\mathcal{D}$ be a class of languages, and A be a language. A is complete for $\mathcal{D}$ under $\leq_m^{\mathcal{D}}$ if A is hard and $A \in \mathcal{D}$.

Notes:

1. Important reducibilities: $\leq_m^P$, $\leq_m^{log}$, $\leq_m^{AC^0}$.

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 $\leq_m^{AC^0}$.

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

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