5n Lower Bound on the Circuit Size

A. Kulikov, O. Melanich, and I. Mihajlin

Steklov Institute of Mathematics at St. Petersburg
Russian Academy of Sciences

Computability in Europe
June 21, 2012
Boolean Circuits

Inputs:
\( x_1, \ldots, x_n, 0, 1 \)

Gates:
binary functions

Fan-out:
unbounded

\[
\begin{align*}
g_1 &= x_1 \oplus x_2 \\
g_2 &= x_2 \land x_3 \\
g_3 &= g_1 \lor g_2 \\
g_4 &= g_2 \lor 1 \\
g_5 &= g_3 \equiv g_4
\end{align*}
\]
Known Lower Bounds

- **Non-constructive**: counting shows that almost all functions of $n$ variables have circuit size $\Theta(2^n/n)$. 
Known Lower Bounds

- Non-constructive: counting shows that almost all functions of $n$ variables have circuit size $\Theta(2^n / n)$.
- Constructive:
Known Lower Bounds

- **Non-constructive:** counting shows that almost all functions of $n$ variables have circuit size $\Theta(2^n/n)$.
- **Constructive:**
  - Full binary basis $B_2$:
    - $3n - o(n)$ [Blum, 1984; Demenkov, Kulikov, 2011]
Known Lower Bounds

- **Non-constructive**: counting shows that almost all functions of \(n\) variables have circuit size \(\Theta(2^n/n)\).

- **Constructive**:
  - Full binary basis \(B_2\):
    \[3n - o(n)\] [Blum, 1984; Demenkov, Kulikov, 2011]
  - Basis \(U_2 = B_2 \setminus \{\oplus, \equiv\}\):
    \[5n - o(n)\] [Iwama, Morizumi, 2002]
This Talk

- This talk: a very simple proof of a $5n - o(n)$ lower bound on the circuit size over $U_2$ for a linear function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\log n}$ (recall that $U_2$ contains all binary functions except for $\oplus$ and $\equiv$).
This Talk

- This talk: a very simple proof of a $5n - o(n)$ lower bound on the circuit size over $U_2$ for a linear function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\log n}$ (recall that $U_2$ contains all binary functions except for $\oplus$ and $\equiv$).
- The proof uses the standard gate elimination technique: during $n - o(n)$ steps we assign a constant to a variable and eliminate at least 5 gates.
This Talk

- This talk: a very simple proof of a $5n - o(n)$ lower bound on the circuit size over $U_2$ for a linear function $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\log n}$ (recall that $U_2$ contains all binary functions except for $\oplus$ and $\equiv$).
- The proof uses the standard gate elimination technique: during $n - o(n)$ steps we assign a constant to a variable and eliminate at least 5 gates.
- The proof consists of four cases only.
Function

- $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\log n}$, $f(x) = Ax$, where $A$ is an $\log n \times n$ matrix s.t. all its columns are pairwise different and non-zero (check-matrix for Hamming codes)
Function

- $f : \{0, 1\}^n \rightarrow \{0, 1\}^{\log n}$, $f(x) = Ax$, where $A$ is an $\log n \times n$ matrix s.t. all its columns are pairwise different and non-zero (check-matrix for Hamming codes)

- important property of $f$: for any two input variables at least one output of $f$ depends essentially on one of these variables only
Proof

- **Case 1**: $\text{out}(x_i) = 1$ — impossible

- **Case 2**: $\text{out}(x_i) \geq 3$ — eliminate 5 gates

- **Case 3**: $\text{out}(x_i) = \text{out}(x_j) = 2$
  - **Case 3.1**: $x_i$ and $x_j$ feed exactly the same two gates — impossible
  - **Case 3.2**: $x_i$ and $x_j$ feed different gates — eliminate 5 gates
Proof

- **Case 1**: $\text{out}(x_i) = 1$ — impossible
- **Case 2**: $\text{out}(x_i) \geq 3$ — eliminate 5 gates

\[ x_i \quad x_j \]

- **Case 3.1**: $x_i$ and $x_j$ feed exactly the same two gates — impossible
- **Case 3.2**: $x_i$ and $x_j$ feed different gates — eliminate 5 gates
Proof

- **Case 1:** $\text{out}(x_i) = 1$ — impossible
- **Case 2:** $\text{out}(x_i) \geq 3$ — eliminate 5 gates
- **Case 3:** $\text{out}(x_i) = \text{out}(x_j) = 2$
Proof

- **Case 1:** $\text{out}(x_i) = 1$ — impossible
- **Case 2:** $\text{out}(x_i) \geq 3$ — eliminate 5 gates
- **Case 3:** $\text{out}(x_i) = \text{out}(x_j) = 2$
  - **Case 3.1:** $x_i$ and $x_j$ feed exactly the same two gates — impossible
Proof

- **Case 1**: \(\text{out}(x_i) = 1\) — impossible
- **Case 2**: \(\text{out}(x_i) \geq 3\) — eliminate 5 gates
- **Case 3**: \(\text{out}(x_i) = \text{out}(x_j) = 2\)
  - **Case 3.1**: \(x_i\) and \(x_j\) feed exactly the same two gates — impossible
  - **Case 3.2**: \(x_i\) and \(x_j\) feed different gates — eliminate 5 gates
Case 1: $\text{out}(x_i) = 1$

assigning the value 0 to $x_j$ makes the circuit independent of $x_i$ while the function must still depend on $x_i$, a contradiction
Case 2: \( \text{out}(x_i) \geq 3 \)

assigning the value 1 to \( x_i \) eliminates at least 5 gates
Case 3.1: out($x_i$) = out($x_j$) = 2 and they feed exactly the same two gates

the circuit does not distinguish between
{$x_i = 0, x_j = 1$} and {$x_i = 1, x_j = 0$}, a contradiction
Case 3.2: \( \text{out}(x_i) = \text{out}(x_j) = 2 \) and they feed different gates

assigning 0 to \( x_j \) eliminates 3 gates and leaves only one wire from \( x_i \)
Thank you for your attention!