An algorithm for producing median formulas for Boolean functions
Reed-Muller Workshop 2011

Miguel Couceiro
Joint work with E. Lehtonen, J.-L. Marichal, T. Waldhauser

University of Luxembourg

May 2011
A Boolean function is a map $f : \{0, 1\}^n \rightarrow \{0, 1\}$, for $n \geq 1$, called the arity of $f$. 
A Boolean function is a map $f : \{0, 1\}^n \to \{0, 1\}$, for $n \geq 1$, called the arity of $f$.

We use the notations $\Omega^{(n)} = \{0, 1\}\{0,1\}^n$ and $\Omega = \bigcup_{n \geq 1} \Omega^{(n)}$. 
A Boolean function is a map $f : \{0, 1\}^n \to \{0, 1\}$, for $n \geq 1$, called the arity of $f$.

We use the notations $\Omega^{(n)} = \{0, 1\}^{\{0,1\}^n}$ and $\Omega = \bigcup_{n \geq 1} \Omega^{(n)}$.

For a fixed arity $n$, the $n$ different projections (variables) $(a_1, \ldots, a_n) \mapsto a_i$ are denoted by $x_1, \ldots, x_n$.

For a fixed arity $n$, the $n$ different negated projections are denoted by $\overline{x_1}, \ldots, \overline{x_n}$. 
A Boolean function is a map $f : \{0, 1\}^n \rightarrow \{0, 1\}$, for $n \geq 1$, called the arity of $f$.

We use the notations $\Omega^{(n)} = \{0, 1\}\{0,1\}^n$ and $\Omega = \bigcup_{n \geq 1} \Omega^{(n)}$.

For a fixed arity $n$, the $n$ different projections (variables) $(a_1, \ldots, a_n) \mapsto a_i$ are denoted by $x_1, \ldots, x_n$.

For a fixed arity $n$, the $n$ different negated projections are denoted by $\overline{x_1}, \ldots, \overline{x_n}$.

For each arity $n$, we denote by

- $\mathbf{0}$ the 0-constant functions.
- $\mathbf{1}$ the 1-constant functions.
The composition of an $n$-ary function $f$ with $m$-ary functions $g_1, \ldots, g_n$ is the $m$-ary Boolean function $f(g_1, \ldots, g_n)$ given by

$$f(g_1, \ldots, g_n)(a) = f(g_1(a), \ldots, g_n(a))$$

for every $a \in \{0, 1\}^m$. 
The composition of an \( n \)-ary function \( f \) with \( m \)-ary functions \( g_1, \ldots, g_n \) is the \( m \)-ary Boolean function \( f(g_1, \ldots, g_n) \) given by

\[
f(g_1, \ldots, g_n)(a) = f(g_1(a), \ldots, g_n(a)) \quad \text{for every} \quad a \in \{0, 1\}^m.
\]

For \( K, J \subseteq \Omega \) the class composition of \( K \) with \( J \), is defined by

\[
K \circ J = \{ f(g_1, \ldots, g_n) : f \text{ \( n \)-ary in } K, \ g_1, \ldots, g_n \text{ \( m \)-ary in } J \}. 
\]
The composition of an $n$-ary function $f$ with $m$-ary functions $g_1, \ldots, g_n$ is the $m$-ary Boolean function $f(g_1, \ldots, g_n)$ given by

$$f(g_1, \ldots, g_n)(a) = f(g_1(a), \ldots, g_n(a))$$

for every $a \in \{0, 1\}^m$.

For $K, J \subseteq \Omega$ the class composition of $K$ with $J$, is defined by

$$K \circ J = \{f(g_1, \ldots, g_n): f \text{ $n$-ary in } K, \ g_1, \ldots, g_n \text{ $m$-ary in } J\}.$$

A (Boolean) clone is a class $C \subseteq \Omega$ containing all projections and satisfying $C \circ C = C$. 
Known results about clones

- Clones constitute an algebraic lattice which was completely classified by Emil Post (1941).
- The class $\Omega$ of all Boolean functions is the largest clone.
- The class $I_c$ of all projections is the smallest clone.
Known results about clones

- Clones constitute an algebraic lattice which was completely classified by Emil Post (1941).
- The class $\Omega$ of all Boolean functions is the largest clone.
- The class $I_c$ of all projections is the smallest clone.
- Each clone $C$ is finitely generated:
  \[
  C = [K], \quad \text{for some finite } K \subseteq \Omega.
  \]
Known results about clones

- Clones constitute an algebraic lattice which was completely classified by Emil Post (1941).

- The class $\Omega$ of all Boolean functions is the largest clone.

- The class $I_c$ of all projections is the smallest clone.

- Each clone $C$ is finitely generated:

  $$ C = [K], \text{ for some finite } K \subseteq \Omega. $$

- Each clone $C$ has a dual clone $C^d = \{f^d : f \in C\}$, where

  $$ f^d(x_1, \ldots, x_n) = f(\overline{x_1}, \ldots, \overline{x_n}). $$
Examples: Essentially Unary Clones

- $I_c = [\cdot]$: Clone of projections.
- $I_0 = [0]$: Clone of projections and 0-constant functions.
- $I_1 = [1]$: Clone of projections and 1-constant functions.
- $I = [0, 1]$: Clone of projections and constant functions.
- $I^* = [\bar{x}]$: Clone of projections and negated projections.
- $\Omega^{(1)} = [0, 1, \bar{x}]$: Clone of essentially unary functions.
We say that $C$ is a **minimal clone** if it covers $I_c$.

- $\Lambda = [\land]$: Clone of conjunctions.
- $\lor = [\lor]$: Clone of disjunctions.
- $L_c = [\oplus_3]$: Clone of constant-preserving linear functions, where $\oplus_3 = x_1 + x_2 + x_3$.
- $SM = [\text{median}]$: Clone of self-dual monotone functions:
  
  $$f = f^d \text{ and } f(a) \leq f(b) \text{ whenever } a \leq b.$$
Known results about composition of clones

- The composition of clones is associative.

- The composition $C_1 \circ C_2$ of clones is not always a clone, e.g., $I^* \circ \Lambda$ is not a clone.

- The composition of clones was completely described by C., Foldes, Lehtonen (2006).

- $\Omega$ can be factorized into a composition of minimal clones.
Descending Irredundant Factorizations of $\Omega$

- **D**: $\Omega = V \circ \Lambda \circ I^*.$

- **C**: $\Omega = \Lambda \circ V \circ I^*.$

- **P**: $\Omega = L_c \circ \Lambda \circ I.$

- **P^d**: $\Omega = L_c \circ V \circ I.$

- **M**: $\Omega = SM \circ \Omega^{(1)}.$
A normal form system (NFS) is a pair \( (\{C_i\}_{1 \leq i \leq k}, \{\gamma_j\}_{1 \leq j \leq k-1}) \) satisfying the following conditions:

- \( \Omega = C_1 \circ \cdots \circ C_{k-1} \circ C_k \), where \( C_k \subseteq \Omega^{(1)} \),

- \( C_i \) is generated by \( \gamma_i \notin C_k \) for \( 1 \leq i \leq k - 1 \),

- \( \gamma_i \neq \gamma_j \) for \( i \neq j \).
An \( n \)-ary formula of a NFS \( (\{C_i\}_{1 \leq i \leq k}, \{\gamma_j\}_{1 \leq j \leq k-1}) \) is a string over \( C_k^{(n)} \cup \{\gamma_j\}_{1 \leq j \leq k-1} \) given by the recursion:

1. The elements of \( C_k^{(n)} \) are \( n \)-ary formulas.

2. If \( \gamma_i \) is \( m \)-ary and \( a_1, \ldots, a_m \) are \( n \)-ary formulas without \( \gamma_j \) for \( i > j \), then \( \gamma_i a_1 \cdots a_m \) is an \( n \)-ary formula.

A formula of a NFS is an \( n \)-ary formula \( \Phi \) for some \( n \), and its length \( |\Phi| \) is the number of symbols occurring in it.
Observe that...
Every $n$-ary formula represents an $n$-ary function, and every $n$-ary function is represented by an $n$-ary formula.

Formulas representing the negation $\overline{x}$:

- $M, D, C$: $\overline{x}$,
- $P, P^d$: $\oplus_3 x01$. 
Let $A$ be a NFS and denote by $F_A$ the set of formulas of $A$.

The $A$-complexity of $f$ is defined by $C_A(f)$, as

$$C_A(f) := \min\{|\Phi| : \Phi \in F_A, \Phi \text{ represents } f\}.$$ 

**$A$-complexities of the negation $\overline{x}$:**

- $C_M(\overline{x}) = C_D(\overline{x}) = C_C(\overline{x}) = 1,$
- $C_P(\overline{x}) = C_{Pd}(\overline{x}) = 4.$
Representations and $A$-complexities of median

Formulas representing median:

$\mathbf{M}$: $\text{median}x_1 x_2 x_3$,

$\mathbf{D}$: $\lor \lor \land x_1 x_2 \land x_1 x_3 \land x_2 x_3$,

$\mathbf{C}$: $\land \land \lor x_1 x_2 \lor x_1 x_3 \lor x_2 x_3$,

$\mathbf{P}$: $\oplus_3 \land x_1 x_2 \land x_1 x_3 \land x_2 x_3$,

$\mathbf{P^d}$: $\oplus_3 \oplus_3 \lor x_1 x_2 \lor x_1 x_3 \lor x_2 x_3 01$. 
Formulas representing median:

\[ M : \text{median} x_1 x_2 x_3, \]
\[ D : \lor \lor \land x_1 x_2 \land x_1 x_3 \land x_2 x_3, \]
\[ C : \land \land \lor x_1 x_2 \lor x_1 x_3 \lor x_2 x_3, \]
\[ P : \bigoplus_3 \land x_1 x_2 \land x_1 x_3 \land x_2 x_3, \]
\[ P^d : \bigoplus_3 \bigoplus_3 \lor x_1 x_2 \lor x_1 x_3 \lor x_2 x_3 01. \]

A-complexities of median:

\[ C_M(\text{median}) = 4, \quad C_D(\text{median}) = C_C(\text{median}) = 11, \]
\[ C_P(\text{median}) = 10, \quad C_{P^d}(\text{median}) = 13. \]
We say that $A$ is \textit{polynomially as efficient as} $B$, denoted $A \preceq B$, if there is a polynomial $p$ with integer coefficients such that

$$C_A(f) \leq p(C_B(f)) \quad \text{for all } f \in \Omega.$$
Comparison of NFSs’

We say that $A$ is **polynomially as efficient as** $B$, denoted $A \preceq B$, if there is a polynomial $p$ with integer coefficients such that

$$C_A(f) \leq p(C_B(f)) \quad \text{for all } f \in \Omega.$$ 

**Fact**

The relation $\preceq$ is a quasi-order on any set of NFSs’.
Comparison of NFSs’

We say that $A$ is **polynomially as efficient as** $B$, denoted $A \preceq B$, if there is a polynomial $p$ with integer coefficients such that

$$C_A(f) \leq p(C_B(f)) \quad \text{for all } f \in \Omega.$$

**Fact**

The relation $\preceq$ is a quasi-order on any set of NFSs’.

If $A \not\preceq B$ and $B \not\preceq A$ holds, then $A$ and $B$ are **incomparable**.

If $A \preceq B$ but $B \not\preceq A$, then $A$ is **polynomially more efficient than** $B$. 
Comparison of NFSs’ (cont.)

Theorem (C., Foldes, Lehtonen)

1. D, C, P, and $P^d$ are incomparable.

2. M is polynomially more efficient than D, C, P, $P^d$. 
A function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is median decomposable if for every $i \in \{1, \ldots, n\}$,

$$f(x) = \text{median}( f(x_i^0), x_i, f(x_i^1) ),$$

where $x_i^c = (x_1, \ldots, x_{i-1}, c, x_{i+1}, \ldots, x_n)$.

**Theorem (Tohma, C., Marichal,...)**

A Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ is monotone iff $f$ is median decomposable.
Algorithm MMNF – Median normal form for monotone Boolean functions

Require: a monotone Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$
Ensure: a median normal form representation of $f$

1: if $n \geq 2$ then
2: $\alpha \leftarrow \text{MMNF}(f(x_1, \ldots, x_{n-1}, 0))$
3: $\beta \leftarrow \text{MMNF}(f(x_1, \ldots, x_{n-1}, 1))$
4: return median $\alpha x_n \beta$
5: else if $f = 0$ then
6: return 0
7: else if $f = 1$ then
8: return 1
9: else
10: return $x_1$
11: end if
Given $f : \{0, 1\}^n \rightarrow \{0, 1\}$, define $g_f : \{0, 1\}^{2n} \rightarrow \{0, 1\}$ as:

for all $b, c \in \{0, 1\}^n$, let

$$g_f(bc) := \begin{cases} 0 & \text{if } \text{weight}(bc) < n, \\ 1 & \text{if } \text{weight}(bc) > n, \\ f(b) & \text{if } b = \overline{c}, \\ 0 & \text{otherwise}. \end{cases}$$
Median representations of arbitrary Boolean functions

Given \( f : \{0, 1\}^n \to \{0, 1\} \), define \( g_f : \{0, 1\}^{2n} \to \{0, 1\} \) as:

for all \( b, c \in \{0, 1\}^n \), let

\[
g_f(bc) := \begin{cases} 
0 & \text{if weight}(bc) < n, \\
1 & \text{if weight}(bc) > n, \\
f(b) & \text{if } b = \overline{c}, \\
0 & \text{otherwise}.
\end{cases}
\]

Facts:

For any Boolean function \( f : \{0, 1\}^n \to \{0, 1\} \),

1. \( g_f \) is monotone;

2. \( f(x_1, \ldots, x_n) = g_f(x_1, \ldots, x_n, \overline{x}_1, \ldots, \overline{x}_n) \).
Algorithm GenMNF – Median normal form for Boolean functions

Require: a Boolean function \( f : \{0, 1\}^n \rightarrow \{0, 1\} \)
Ensure: a median normal form representation of \( f \)

1: \textbf{if} \( f \) is monotone \textbf{then}
2: \quad \textbf{return} \ MMNF(f)
3: \textbf{else}
4: \quad \text{Construct} \ g_f \text{ as shown.}
5: \quad w \leftarrow \text{MMNF}(g_f)
6: \quad \textbf{for} i = 1 \text{ to } n \textbf{do}
7: \quad \quad \text{Replace each occurrence of } x_{n+i} \text{ in } w \text{ by } \overline{x_i}.
8: \quad \textbf{end for}
9: \quad \textbf{return} \ w
10: \textbf{end if}
Thank you for your attention!