The Complexity of Mind Changes

Bernd Borchert

Abstract

The notion of the maximal number of mind changes for a Boolean function was defined and applied in several contexts. An application in complexity theory is the result of Wagner and Wechsung that the classes of the Boolean closure of NP are exactly the classes of the Boolean hierarchy over NP. The aim of this paper is to study the complexity of determining the maximal number of mind changes of a Boolean function if the function is represented as a circuit.

1 Introduction

The maximal number of mind changes of a Boolean function f is the maximal number of changes of the value of f on the ascending sequences of assignments, an exact definition is given in the next section. This invariant for a Boolean function has been defined explicitly (with other names) or ˚implicitly in several papers, for example in [Gil54, Mar58, Hay78, WW85, KSW86, Cha91], this is probably not a full list. To motivate the interest in the notion, consider the result of Wagner and Wechsung in [WW85] obtained with the help of the maximal number of mind changes invariant:

Take any class $\mathcal{A}$ of languages and a Boolean function $f(x_1, \ldots, x_n)$ then $f(\mathcal{A})$ is defined to be the class of languages $f(L_1, \ldots, L_n)$ for all combinations $L_1, \ldots, L_n$ of languages in $\mathcal{A}$, where $f(L_1, \ldots, L_n)$ is the language L whose characteristic function χ_L is equal to $f(\chi_{L_1}, \ldots, \chi_{L_n})$. Given a class $\mathcal{A}$, consider the set of classes $\{f(\mathcal{A}) \mid f \text{ is a Boolean function}\}$, and ask for which pairs of classes inclusion resp. equality holds. Generally, this gives a messy picture. The surprising result of

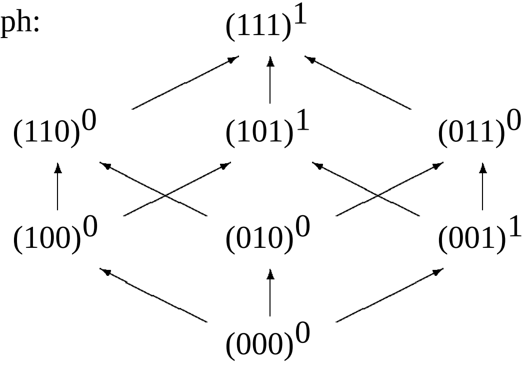
[WW85] states that for the class NP the classes f(NP) are ordered nearly linearly and are (for non–constant f's) exactly the classes of the Boolean hierarchy defined in [CGH*88]. Actually, it is just this result of [WW85] which should allow to call the Boolean hierarchy this way.

The aim of this paper is to investigate the complexity to determine the maximal number of mind changes of a Boolean function if the function is given as a circuit. For example the following problem will be shown for every $n \geq 2$ to be DP–complete: given a circuit f, is n the maximal number of mind changes of the Boolean function computed by f?

2 Preliminaries

Let $V = \{x_1, \ldots, x_n\}$ be a set of n variables. Let $\mathcal{A}_V$ be the set of assignments for V, i.e. the set of all mappings from V to $\{0, 1\}$. Note that $\mathcal{A}_V$ is the set of characteristic functions χ_W of subsets $W \subseteq V$. For V an order on the variables is fixed and the assignments are written as vectors $(0 \ldots 0), \ldots, (1 \ldots 1)$ listing the values of the characteristic function. Consider the partial order $(\mathcal{A}_V, \subseteq)$ (which is in fact a Boolean algebra) induced by set inclusion on V: for $a, a' \in \mathcal{A}_V$ define $a \subseteq a'$ iff $a = \chi_W, a' = \chi_{W'}$ and $W \subseteq W'$. Define $a \subset a'$ iff $a \subseteq a'$ and $a \neq a'$.

A *Boolean function* f on a set V of variables is a mapping from $\mathcal{A}_V$ to $\{0, 1\}$. In this paper it is helpful to imagine a Boolean function $f(V)$ to be represented by the graph of $(\mathcal{A}_V, \subseteq)$ where only direct successors are marked with arrows and where each assignment a is labeled with the value $f(a)$. For example, the Boolean function defined by the formula $(x_1 \vee \overline{x_2}) \wedge x_3)$ is represented by the following graph:



For $V = \{x_1, \ldots, x_n\}$ an *ascending sequence* in $(\mathcal{A}_V, \subseteq)$ is defined to be a sequence of assignments $a_1, \dots, a_m$ such that $a_i \subset a_{i+1}$ for $i = 1, \dots, m-1$. For a Boolean function f on V and an ascending sequence $a_1, \dots, a_m$ in $(\mathcal{A}_V, \subseteq)$ consider the sequence $f(a_1), \dots, f(a_m)$. Call the number of i's in $\{1, \dots, m-1\}$ such that $f(a_i) \neq f(a_{i+1})$ the *number of mind changes of f on the ascending sequence $a_1, \dots, a_m$*. For a Boolean function f on V let the *maximal number of mind changes of f*, written **mc**(f), be the maximal number of mind changes of f on an ascending sequence in $(\mathcal{A}_V, \subseteq)$. Note that a Boolean function is constant iff the maximal number of mind changes is 0. Refine the notation by saying that the maximal number of mind changes of f is s^- (s^+) iff the maximal number of mind changes of f is s and $f(0 \dots 0) = 0$ ($f(0 \dots 0) = 1$). For example the maximal number of mind changes of the function in the picture above is 3^-.

3 Applications of the mind change notion

As an application of the notion of maximal number of mind changes the result of [WW85]°will be stated, for the formulation the following definitions are necessary: NP is the set of languages which are accepted by a polynomial time nondeterministic machine, NP(s) and co–NP(s) for $s \in N$ are the classes of the Boolean hierarchy in the notation of [CGH*88]. For a Boolean function f the class f(NP) was defined in the introduction.

The following theorem is from [WW85]. Actually, it seems to follow as a corollary from the results of [Hay78].

Theorem 1 (Wagner/Wechsung) *Let f be a non–constant Boolean function.*
(a) If **mc**$(f) = s^-$ *then* f(NP) = NP(s).
(b) If **mc**$(f) = s^+$ *then* f(NP) = co–NP(s).

The case that f is constant is left out because in that case f(NP) is either $\{\emptyset\}$ or $\{\Sigma^*\}$ but not NP(0) = co–NP(0) =P.

The results of Hausdorff [Hau14] only imply the following weaker version of the previous theorem: for every Boolean function f there is an s such that f(NP) $\subseteq$ NP(s). This was stated in [CGH*88].

Because the Boolean hierarchy for the class RE of the recursively enumerable sets is known to be proper one can state for RE with the notation of [Ers68] an even stronger analogon of the previous theorem. It follows as a corollary from the results of Hay [Hay78] and Selivanov [Sel84] but also could be derived by the methods of Wagner and Wechsung [WW85].

Theorem 2 (Hay, Selivanov, Wagner/Wechsung) *Let f be a non–constant Boolean function.*
(a) **mc**$(f) = s^-$ $\iff$ $f(\mathrm{RE}) = \Sigma_s^{-1}$.
(b) **mc**$(f) = s^+$ $\iff$ $f(\mathrm{RE}) = \Pi_s^{-1}$.

In [KSW86] it was already remarked that an analogous theorem like the two ones above is valid generally for every class closed under union and intersection, not only for NP and RE.

Another application of the mind change notion in complexity theory is the following. For a Boolean function f the *inversion complexity* is the minimal number of negations gates one needs in a circuit to represent f. This corresponds to a complexity measure for circuits where conjunction gates, disjunction gates, variables, constants and wires are for free but each negation gate has cost 1.

The following result is an example of an optimal complexity bound, the upper bound part was given in [Gil54] and the lower bound part was added in [Mar58].

Theorem 3 (Gilbert, Markov) *Let f be a non–constant Boolean function. If the maximal number of mind changes of f is $(s+1)^-$ or s^+ then the inversion complexity of f is the smallest number n such that $s < 2^n$.*

4 Complexity questions

In the previous section it was indicated that the maximal number of mind changes may be an interesting invariant of a Boolean function. The aim of this paper is to study the complexity to determine this number if the Boolean function is represented in a short way, i.e. as a formula or a circuit. All results will be stated for circuits but are also valid for formulas.

Assume that Boolean circuits are encoded in a straightforward way as strings over some alphabet, let $\mathcal{C}$ be the set of correct codes of circuits. Note that each Boolean circuit computes a Boolean function on the set of the occuring variables, call the Boolean function computed by a circuit f also f, this should not cause confusion.

Problems which are polynomial time many–one complete for a class $\mathcal{A}$ will be just called $\mathcal{A}$*–complete*. Remember that SAT = $\{f \in \mathcal{C} \mid f(a) = 1$ for some assignment $a\}$ is NP-complete. For two languages A, B write $A \equiv_m^p B$ iff A and B are polynomial time many–one equivalent. Let for two languages A, B $A \oplus B$ be the language $0A \cup 1B$.

Define for every $n \geq 1$ the circuit $\psi_n(x_1, \ldots, x_n)$ where all x_i are different variables by $\psi_1(x_1) := x_1$ and $\psi_{n+1}(x_1, \ldots, x_{n+1}) := x_1 \wedge \neg\psi_n(x_2, \ldots, x_{n+1})$. Note that **mc**($\psi_n$) = n^- because $\psi_n(0 \ldots 0) = 0$ and ψ_n has on the ascending sequence $(00 \ldots 0), (10 \ldots 0), (11 \ldots 0), \ldots, (11 \ldots 1)$ n mind changes. Therefore the maximal number of mind changes of $\neg\psi_n$ is n^+. Like in [CGH*88] let NP(0) = P, NP(1) = NP and let for $n \geq 1$ NP(n + 1) be the set of intersections of a language in NP and the complement of a language in NP(n), i.e. NP(n + 1) = ψ_{n+1}(NP). Let co–NP(n) be the set of complements of languages in NP(n). Remember that NP(2) = DP. Let $\langle \ldots \rangle$ be a pairing function. Define for $n \geq 1$ the language SAT(n) like in [CGH*88] by SAT(1) = SAT and SAT(n + 1) = $\{\langle f, g\rangle \mid f \in$ SAT$, g \notin$ SAT$(n)\}$, and let co–SAT(n) be the complement of SAT(n). In [CGH*88] it was show that SAT(n) is NP(n)–complete and that co–SAT(n) is co–NP(n)–complete.

For special subsets S of N the computational complexity of the following problem will be studied:

MC^S := $\{f \in \mathcal{C} \mid$ the maximal number of mind changes of f is in $S\} \cup \{\epsilon\}$

ϵ is added in order to guarantee that **MC**^{S} is never empty. First a theorem about the complexity of **MC**^{S} for quite general sets S will be proved and then the complexities of more natural problems like for example **MC**$^{\{n\}}$ will follow as corollaries.

For a subset S of N let S_{even} be the set of even numbers in S and S_{odd} the set of odd numbers in S, therefore S is the disjunct union $S_{\text{even}} \cup S_{\text{odd}}$. Let S' be either S_{even} or S_{odd}. Say that S' *has* s *alternations* iff s is the cardinality of the set $\{i \in N \mid (i \in S', i + 2 \notin S')$ or $(i \notin S', i + 2 \in S')\}$. Note that S' has a finite number of alternations iff S' is finite or cofinite. Refine the notation by saying that S_{even} has s^- (s^+) alternations iff $0 \notin S_{\text{even}}$ ($0 \in S_{\text{odd}}$), and that S_{odd} has s^- (s^+) alternations iff $1 \notin S_{\text{even}}$ ($1 \in S_{\text{odd}}$).

Here is the main theorem, all other results will follow as corollaries.

Theorem 4 *Let* S *be a subset of* N *such that both* S_{even} *and* S_{odd} *have a finite number of alternations. Let* S' *be either* S_{even} *or* S_{odd}*.*

(a) If S' *has* s^- *alternations then* **MC**$^{S'}$ *is* NP(s)*–complete.*

(b) If S' has s^+ alternations then $\mathbf{MC}^{S'}$ *is* co–NP(s)*–complete.*
(c) $\mathbf{MC}^{S} \equiv^p_m \mathbf{MC}^{S_{\text{even}}} \oplus \mathbf{MC}^{S_{\text{odd}}}$.

Proof: (a) Let $S' = S_{\text{even}}$ and let S' have n^- alternations. If $n = 0$ then $S' = \emptyset$ and $\mathbf{MC}^{S'} = \{\epsilon\}$ which is NP(0) = P–complete. For $n \geq 1$ the proof will be done by induction on n, and a polynomial-time computable function $\Phi_{S'}$ for a many-one reduction from SAT(n) to $\mathbf{MC}^{S'}$ will be defined inductively.

Let n be ≥ 1 and let $S' = S_{\text{even}}$ have n^- alternations, i.e. $0 \notin S'$. Let $a > 0$ be the smallest number such that $a \in S'$, and define $a + 2N$ to be the set $\{a + 2m \mid m \in N\}$. Consider the following nondeterministic program to recognize $\mathbf{MC}^{a+2N}$: given a circuit f on n variables guess a sequence of $n + 1$ assignments and accept if the sequence is ascending and $f(0\dots0) = f(1\dots1)$ and f has at least a mind changes on the ascending sequence. This program accepts $\mathbf{MC}^{a+2N}$ nondeterministically in quadratic time, therefore $\mathbf{MC}^{a+2N} \in$ NP. If $n = 1$ then $S' = a + 2N$ and therefore $\mathbf{MC}^{S'} \in$ NP(1). If $n \geq 2$ let X be the set of even numbers $\geq a$ which are not in S'. Obviously X has $(n-1)^-$ alternations and by induction is $\mathbf{MC}^{X}$ in NP($n-1$). S' is the intersection of $a + 2N$ and the complement of X and therefore $\mathbf{MC}^{S'}$ is in NP(n) because it is a intersection of a language in NP and the complement of a language in NP($n-1$). It remains to show the completeness, this will be done by a polynomial-time reduction of SAT(n) to $\mathbf{MC}^{S'}$.

If $n = 1$ then $S' = a + 2N$ and the following function $\Phi_{S'}$ is a many-one reduction from SAT(1) = SAT to $\mathbf{MC}^{a+2N}$: given a circuit $f(x_1, \dots, x_k)$ check if $f(0\dots0) = f(1\dots1) = 0$; if not then map f to ψ_a; if yes then map f to $\psi_{a-2}(y_1, \dots, y_{a-2}) \vee f(x_1, \dots, x_k)$ where $y_1, \dots, y_{a-2}$ are chosen to be different from the variables $x_1, \dots, x_k$. This is a correct reduction: if f is not satisfiable then the maximal number of mind changes of $\Phi_{S'}(f)$ is $a-2$ and therefore $\mathbf{mc}(\Phi_{S'}(f)) \notin S'$, and if f is satisfiable then $\Phi_{S'}(f)$ has at least a mind changes.

Consider the Boolean function on $2k$ variables which outputs 1 for an assignment a iff a maps exactly k of the variables to 1. According to Theorem 4.1 of [Weg87] there is for every k a circuit $H_k(x_1, \dots, x_{2k})$ of size polynomial in k which computes this Boolean function. From the proof for the theorem it can be observed that the construction of H_k from k is possible in polynomial time in k.

Let Y be the set of even numbers i such that $i + a$ is not in S'. Note that Y has $(n-1)^-$ alternations. By induction there exists a polynomial-time computable function Φ_Y such that $t \in$ SAT($n-1$) $\iff$ $\Phi_Y(t)$ is a circuit whose maximal number of mind changes is in Y.

Given a pair $q = \langle f(x_1, \dots, x_k), t\rangle$ define

$\Phi_{S'}(q) := \psi_{a-2}(y_1, \dots, y_{a-2}) \vee [g(x_1, x_1', \dots, x_k, x_k') \wedge \neg\Phi_Y(t)]$

with $g(x_1, x'_1, \dots, x_k, x'_k) := f(x_1, \dots, x_k) \wedge H_k(x_1, x'_1, \dots, x_k, x'_k)$

where the variables y_i, x_i, x'_i are chosen resp. assumed to be different from each other and different from the variables in $\Phi_Y(t)$.

It remains to show the correctness of the reduction: $[f(x_1, \dots, x_k) \in$ SAT and $t \notin$ SAT$(n-1)]$ $\iff$ $\Phi_{S'}(q)$ is a circuit whose maximal number of mind changes is in S'.

If f is not satisfiable then $\Phi_{S'}$ is equivalent to ψ_{a-2} whose maximal number of mind changes is $a-2$ which is not in S'. It remains to show that if f is satisfiable then: $t \notin$ SAT$(n-1)$ $\iff$ $\Phi_{S'}(q)$ is a circuit whose maximal number of mind changes is in S'.

If f is satisfiable by some assignment a for the variables $\{x_1, \dots, x_k\}$ then there is an extension to an assignment a' for the variables $\{x_1, x'_1, \dots, x_k, x'_k\}$ which maps half of the variables $x_1, \dots, x_k, x'_1, \dots, x'_k$ to 1 and therefore evaluates g to 1. But all assignments for which $H_k(x_1, x'_1, \dots, x_k, x'_k)$ evaluates to 1 are $\subseteq$–incomparable, and $g(0\dots0) = g(1\dots1) = 0$. Thus if $f \in$ SAT then the maximal number of mind changes of g is 2 and therefore **mc**$(\Phi_{S'}(q)) = a +$ **mc**$(\Phi_Y(t))$. But then: $t \notin$ SAT$(n-1)$ $\iff$ (by induction hypothesis) **mc**$(\Phi_Y(t)) \notin Y$ $\iff$ (by defintion of Y) **mc**$(\Phi_Y(t)) + a \in S'$ $\iff$ (by the equation above) **mc**$(\Phi_{S'}(q)) \in S'$. This finishes the correctness proof of the given reduction.

For $S' = S_{\text{odd}}$ the proof is analog.

(b) The proof is dual to (a).

(c) Note that for each circuit $f(x_1, \dots, x_k)$: $f(0\dots0) = f(1\dots1)$ iff the maximal number of mind changes of f is even.

q.e.d.

Note that the problems $\mathbf{MC}^S$ for sets S like in the theorem above are therefore a collection of complete problems for the classes of the Boolean Hierarchy and the Query Hierarchy, see [Bei91, Cha91].

One gets the following corollaries for some natural sets S:

Corollary 1 *For each* $n \geq 1$ $\mathbf{MC}^{\{n,n+1,n+2,\dots\}}$ *is* NP-*complete.*

Corollary 2 *For each* $n \in N$ $\mathbf{MC}^{\{0,\dots,n\}}$ *is* co–NP-*complete.*

Corollary 3 $\mathbf{MC}^{\{0\}}$ *and* $\mathbf{MC}^{\{1\}}$ *are* co–NP-*complete while for every* $n \geq 2$ $\mathbf{MC}^{\{n\}}$ *is* DP–*complete.*

This implies for $n = 1, 2$ together with Theorem 1 the following slightly confusing statements:

Corollary 4 *Assume that the Boolean hierarchy does not collapse.*
(a) Then the problem $\{f \in \mathcal{C} \mid f(\mathrm{NP}) = \text{co–NP}\}$ *is* co–NP–*complete.*
(b) Then the problem $\{f \in \mathcal{C} \mid f(\mathrm{NP}) = \mathrm{DP}\}$ *is* DP–*complete.*

5 Acknowledgements

I am grateful to Richard Chang for explaining me the mind change notion and to Victor Selivanov for asking me the question how difficult it is to determine the maximal number of mind changes of a given circuit.

References

[Bei91] R.Beigel *Bounded queries to SAT and the Boolean Hierarchy*, Theoretical Computer Science **84**, 1991, pp. 199–223

[CGH*88] J.Y.Cai, Th.Gundermann, J.Hartmanis, L.Hemachandra, V.Sewelson, K.Wagner, G.Wechsung. *The Boolean Hierarchy 1: Structural Properties*, SIAM Journal of Computing **17**, No. 6, 1988, pp. 1232–1252

[Cha91] R.Chang. *On the Structure of NP Computations under Boolean Operators*, Ph.D Thesis, Cornell University, 1991

[Ers68] Y.L.Ershov. *A Hierarchy of Sets I*, Algebra and Logic **7**, 1968, pp. 25–43

[Gil54] E.N.Gilbert. *Lattice Theoretic Properties of Frontal Switching Functions*, Journal of Mathematical Physics **33** (now Journal of Applied Mathematics), 1954, pp. 57–67

[Hau14] F.Hausdorff. *Grundzüge der Mengenlehre*, Leipzig, 1914

[Hay78] L.Hay. *Convex Subsets of* 2^n *and Bounded Truth-Table Reducibility*, Discrete Mathematics **21**, 1978, pp. 31–46

[KSW86] J.Köbler, U.Schöning, K.W.Wagner *The Difference and Truth–Table Hierarchies for* NP, Fachberichte Informatik, EWH Koblenz, Bericht 7/86, 1986

[Mar58] A.A.Markov. *On the Inversion Complexity of a system of functions*, Journal of the ACM **5**, 1958, pp. 331–334

[Sel84] V.Selivanov. *Index Sets in the Hyperarithmetical Hierarchy*, Siberian Mathematical Journal **25** (English translation), 1984, pp. 474–488

[WW85] G.Wechsung. *On the Boolean Closure of NP*, FCT Conference 1985, Springer Verlag, LNCS 199, 1985, pp. 485–493 (an unpublished precursor of this paper was coauthored by K.Wagner)

[Weg87] I.Wegener. *The Complexity of Boolean Functions*, Teubner, Stuttgart, 1987