On the Relative Complexity of Post's Classes

Elmar Böhler

Lehrstuhl Informatik IV
Bayerische Julius-Maximilians-Universität Würzburg
Am Hubland
97074 Würzburg

Abstract

We discuss how difficult it is to decide whether or not a Boolean formula or a Boolean circuit is from a class of Post's. We consider the problem of deciding a class of Post's if the decision algorithm has the assurance / the promise that it's inputs will be of another class of Post's. We show for nearly all possible subject/promise combinations of Post's classes, that the resulting problems are either in P or coNP-complete.

Keywords: propositional logic, computational complexity, promises

1 Introduction

In computer science Boolean logic is of central interest. It is the foundation for such different fields as databases, layout of integrated circuits, communication, or artificial intelligence. In complexity theory, formula- and circuit-based combinatorial problems were researched in detail and in many cases when a new complexity class was introduced, a problem related to Boolean logic was the first one for which the completeness was proven. So S. A. Cook, for example, proved that the satisfiability problem for Boolean formulas is NP-complete [Coo71]: It was the first NP-complete set ever discovered.
Already in the twenties of the last century E. L. Post studied in depth sets of Boolean functions and their closures under *superposition* i.e. introduction of fictive variables, identification of variables, permutation of variables and substitution - these are the basic operations used when constructing Boolean circuits.
Post identified all closures of Boolean functions under superposition, characterized their inclusion structure and fixed a finite base for each of them (see [Pos41], [JGK70], Figure 1 and Table 1). In 1979 Lewis could make a more precise statement about the satisfiability of propositional formulas. He could show in a dichotomy theorem, that depending on the connectives used to build formulas, the satisfiability problem is either NP-complete or in P. Only recently further investigations exploiting Post's result were done by Reith, Vollmer and Wagner. In 1999 Reith and Wagner studied in more detail so called B-formulas i.e. formulas that are in Lewis's sense build over connectives from a defined set B of Boolean functions. Reith and Wagner gave a complete characterization of the circuit value problem and the satisfiability problem for B-circuits in [RW00]. In [RV00] Reith and Vollmer researched the complexity of finding the minimal and maximal satisfying assignment for B-formulas.
In this report we provide an answer to the question of how difficult Post's classes are to decide i.e. how difficult it is for a given $\{et, vel, non\}$-circuit to decide whether or not it is from a given class of Post's. We call this the membership problem for Post's classes.

To choose $\{et, vel, non\}$-circuits is straightforward since every Boolean function can be described by such a circuit or, with other words, the closure of $\{et, vel, non\}$ under superposition is the set of all Boolean functions. But how about the membership problem for circuits that are built over a different set of Boolean functions? Surely there are sets B such that the membership problems for B-circuits are easier to decide than they are for $\{et, vel, non\}$-circuits.
Depending on the choice of the set B of Boolean functions we utilize to build circuits, the set A of Boolean functions that can be described varies. Those sets A coincide with Post's classes, and an algorithm that has to decide the membership problem for B-circuits will take advantage mostly of the fact that the Boolean function described by a B-circuit is from A.
We therefore focus the attention of this paper not on the membership problem for B-circuits but on the membership problem for $\{et, vel, non\}$-circuits, but we provide for an algorithm that has to decide whether or not an $\{et, vel, non\}$-circuit is from a class of Post's K the information that the circuit is from another class of Post's K'. With other words, such an algorithm can rely on the promise that all its inputs are circuits describing Boolean functions from K', independently of the base utilized to build the circuit.
This approach does not only deliver an upper bound for membership problems for B-circuits, it also makes us sensible for the amount of information inherent in a statement like "a Boolean function is linear" and we get a good impression about the complexity relationship of Post's classes.

2 Preliminaries

2.1 Boolean Functions, Superposition, B-Circuits

For a set B of Boolean functions let $[B]$ be the smallest set of Boolean functions that contains B and is closed under permutation and identification of variables, under introduction of fictive variables and under substitution (i.e. composition of functions). We subsume these four operations under the term *superposition*. We say "closed classes of Boolean functions" or simply "closed classes" when we mean "classes of Boolean functions being closed under superposition" for the sake of abbreviation whenever the context is non-ambiguous.
A set A of Boolean functions is called *base* of B if $[A] = B$ and A is called *complete* if $[A] = \mathrm{BF}$, i.e. if the closure of A under superposition is the set of all Boolean functions.

Definition 1.
Let f^n be a Boolean function of arity n.

- *f is called* a-reproducing *if $f(a, \ldots, a) = a$ for $a \in \{0, 1\}$.*
- *We define a relation $\leq$ on $\{0,1\}^n \times \{0,1\}^n$ with $(a_1, \ldots, a_n) \leq (b_1, \ldots, b_n)$ iff $a_i \leq b_i$ holds for all $i \in \{1, \ldots, n\}$. f^n is called* monotonic *if $\alpha \leq \beta$ implies $f^n(\alpha) \leq f^n(\beta)$ for all $\alpha, \beta \in \{0,1\}^n$.*
- *Let $T \subseteq \{0,1\}^n$ and $a \in \{0,1\}$. We call T* a-separating *if there exists an $i \in \{1, \ldots, n\}$ such that for all $(b_1, \ldots, b_n) \in T$ holds $b_i = a$.*

- *f is called* a-separating *if $f^{-1}(a)$ is a-separating.*
- *f is called* a-separating of degree m *if every $T \subseteq f^{-1}(a)$ with $|T| = m$ is a-separating.*
- *Let* $\mathrm{dual}(f^n)(a_1, \ldots, a_n) =_{\mathrm{def}} \overline{f^n(\overline{a_1}, \ldots, \overline{a_n})}$ *for all $a_1, \ldots, a_n \in \{0, 1\}$. Observe that* $\mathrm{dual}(\mathrm{dual}(f)) = f$ *holds.*
- *f^n is called* self-dual *if* $f^n = \mathrm{dual}(f^n)$.
- *Let A be a set of Boolean functions. Then* $\mathrm{dual}(A) =_{\mathrm{def}} \{\mathrm{dual}(f) | f \in A\}$.
- *For $1 \leq i \leq n$ the i-th variable is called* fictive *if* $f(a_1, \ldots, a_{i-1}, 0, a_{i+1}, \ldots, a_n) = f(a_1, \ldots, a_{i-1}, 1, a_{i+1}, \ldots, a_n)$ *for all $a_1, \ldots, a_n \in \{0, 1\}$. A variable that is not fictive is called* relevant.
- *We denote the set of all variables of f^n with* $\mathrm{var}(f^n)$ *and the set of all relevant variables of f^n with* $\underline{\mathrm{var}}(f^n)$.

We refer to important Boolean functions with the latin terms *et*, *vel*, *non*, *id*, *aut*, *seq* and *aeq*. Constant functions will often be referred to with 0 and 1. Furthermore we denote for some integer n and $m \in \{1, \ldots, n\}$ the n-ary identity at position m with I^n_m and the n-ary negation at position m with N^n_m. If not stated otherwise we will denote tuples of binary values with greek letters and for an $\alpha = (\alpha_1, \ldots, \alpha_n)$ from $\{0, 1\}^n$ we write $\overline{\alpha}$ for $(\overline{\alpha_1}, \ldots, \overline{\alpha_n})$.

Proposition 2. ([JGK70]) *Let g^m be a Boolean function with $g(\alpha) = f(f_1(\alpha), \ldots, f_n(\alpha))$ for $\alpha \in \{0, 1\}^m$ and $f, f_1, \ldots, f_n \in \mathrm{BF}$. Then* $\mathrm{dual}(g(\alpha)) = \mathrm{dual}(f)(\mathrm{dual}(f_1)(\alpha), \ldots, \mathrm{dual}(f_n)(\alpha))$.

We need a means to describe Boolean functions on two levels: We will on the one hand have objects that represent Boolean functions and can be used as input argument for algorithms: We choose circuits for that. And on the other hand Boolean functions will have to be described in the text: This will be done using propositional formulas. We will provide a definition of *circuits* now, that is derived from the one given in [Vol99] where one can find a more general introduction to Boolean circuits.

Definition 3. *Let B be a finite set of Boolean functions. A B-*circuit *is a tuple*

$$C = (V, E, \alpha, \beta, o)$$

where (V, E) is a finite directed acyclic graph, $\alpha : E \mapsto \mathbb{N}$ is an injective function, $\beta : V \mapsto B \cup \{x_1, \ldots, x_n\}$ and $o \in V$, such that the following conditions hold:

- *If $v \in V$ has in-degree 0, then $\beta(v) \in \{x_1, \ldots, x_n\}$ or $\beta(v)$ is a 0-ary Boolean function from B.*
- *If $v \in V$ has in-degree $k > 0$, then $\beta(v)$ is a k-ary Boolean function from B.*

α is introduced since the input order of non-commutative Boolean functions has to be defined non-ambiguously. We define a semantics for B-circuits by giving the function $val : V \times \{0, 1\}^n \mapsto \{0, 1\}$ that maps for every $\{0, 1\}^n$-tuple every vertex in the graph to a binary value as follows:

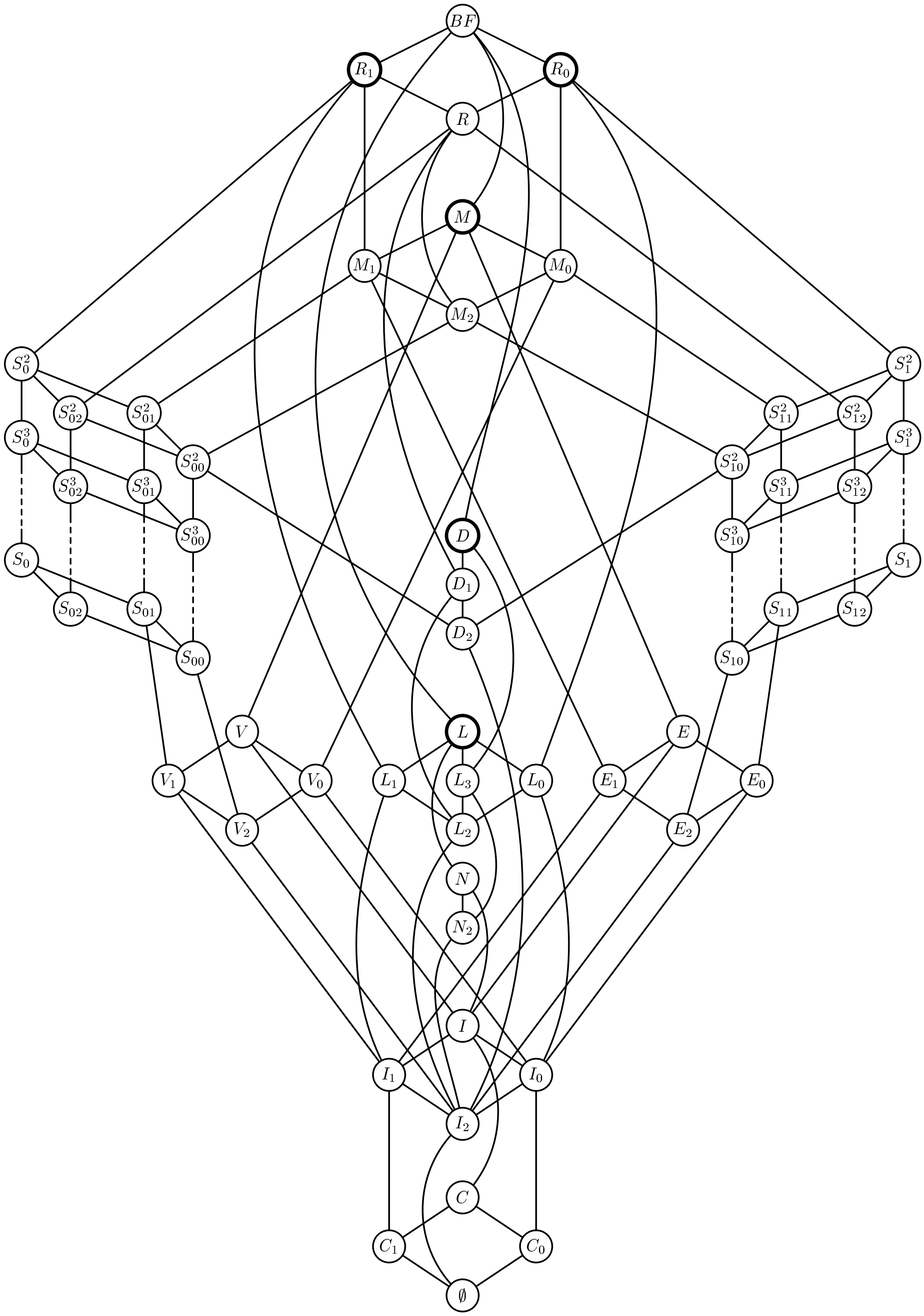

Figure 1: Graph of all classes of Boolean functions being closed under superposition

Table 1: List of all classes of Boolean functions with examples of bases

Name	**Definition**	**Base(s)**
BF	all Boolean functions	{et, non},{vel, non},{sh},{nd}
R_0	$\{f \in BF \| f$ is 0-reproducing$\}$	{et, aut}
R_1	$\{f \in BF \| f$ is 1-reproducing$\}$	{vel, $x \oplus y \oplus c_1$}
R	$R_1 \cap R_0$	{vel, $x \wedge (y \oplus z \oplus c_1)$}
M	$\{f \in BF \| f$ is monotonic$\}$	{et, vel, c_0,c_1}
M_1	$M \cap R_1$	{et, vel, c_1}
M_0	$M \cap R_0$	{et, vel, c_0}
M_2	$M \cap R$	{et, vel}
S_0^n	$\{f \in BF \| f$ is 0-separating of degree n$\}$	{seq, dual(h_n)}
S_0	$\{f \in BF \| f$ is 0-separating$\}$	{seq}
S_1^n	$\{f \in BF \| f$ is 1-separating of degree n$\}$	$\{x \wedge \overline{y}, h_n\}$
S_1	$\{f \in BF \| f$ is 1-separating$\}$	$\{x \wedge \overline{y}\}$
S_{02}^n	$S_0^n \cap R$	{$x \vee (y \wedge \overline{z})$, dual($h_n$)}
S_{02}	$S_0 \cap R$	$\{x \vee (y \wedge \overline{z})\}$
S_{01}^n	$S_0^n \cap M$	{dual(h_n), c_1}
S_{01}	$S_0 \cap M$	$\{x \vee (y \wedge z), c_1\}$
S_{00}^n	$S_0^n \cap R \cap M$	{$x \vee (y \wedge z)$, dual(h_n)}
S_{00}	$S_0 \cap R \cap M$	$\{x \vee (y \wedge z)\}$
S_{12}^n	$S_1^n \cap R$	$\{x \wedge (y, \overline{z}), h_n\}$
S_{12}	$S_1 \cap R$	$\{x \wedge (y \vee \overline{z}\}$
S_{11}^n	$S_1^n \cap M$	$\{h_n, c_0\}$
S_{11}	$S_1 \cap M$	$\{x \wedge (y \vee z), c_0\}$
S_{10}^n	$S_1^n \cap R \cap M$	$\{x \wedge (y \vee z), h_n\}$
S_{10}	$S_1 \cap R \cap M$	$\{x \wedge (y \vee z)\}$
D	$\{f \| f$ is self-dual$\}$	$\{x\overline{y} \vee x\overline{z} \vee \overline{y} \wedge \overline{z}\}$
D_1	$D \cap R$	$\{xy \vee x\overline{z} \vee y\overline{z}\}$
D_2	$D \cap M$	$\{xy \vee yz \vee xz\}$
L	$\{f^n \in BF \|$ There exists a formula of the form $c_0 \oplus c_1x_1 \oplus \cdots \oplus c_nx_n$ where c_i are constants for all $0 \leq i \leq n$, that describes $f^n\}$	{aut, c_1}
L_0	$L \cap R_0$	{aut}
L_1	$L \cap R_1$	{aeq}
L_2	$L \cap R$	$\{x \oplus y \oplus z\}$
L_3	$L \cap D$	$\{x \oplus y \oplus z \oplus c_1\}$
V	$\{f \|$ There exists a formula of the form $c_0 \vee c_1x_1 \vee \cdots \vee c_nx_n$ where c_i are constants for all $0 \leq i \leq n$, that describes $f^n\}$	{vel, c_0, c_1}
V_0	$[\{vel\}] \cup [\{c_0\}]$	{vel, c_0}
V_1	$[\{vel\}] \cup [\{c_1\}]$	{vel, c_1}
V_2	$[\{vel\}]$	{vel}
E	$\{f \|$ There exists a formula of the form $c_0 \wedge (c_1 \vee x_1) \wedge \cdots \wedge (c_n \vee x_n)$ where c_i are constants for all $0 \leq i \leq n$, that describes $f^n\}$	{et, c_0, c_1}
E_0	$[\{et\}] \cup [\{c_0\}]$	{et, c_0}
E_1	$[\{et\}] \cup [\{c_1\}]$	{et, c_1}
E_2	$[\{et\}]$	{et}
N	$[\{non\}] \cup [\{c_0\}] \cup [\{c_1\}]$	{non, c_1}, {non, c_0}
N_2	$[\{non\}]$	{non}
I	$[\{id\}] \cup [\{c_1\}] \cup [\{c_0\}]$	{id, c_0, c_1}
I_0	$[\{id\}] \cup [\{c_0\}]$	{id, c_0}
I_1	$[\{id\}] \cup [\{c_1\}]$	{id, c_1}
I_2	$[\{id\}]$	{id}
C	$[\{c_1\}] \cup [\{c_0\}]$	$\{c_1, c_0\}$
C_0	$[\{c_0\}]$	$\{c_0\}$
C_1	$[\{c_1\}]$	$\{c_1\}$

$(h_n =_{\text{def}} \bigvee_{i=1}^{n+1} x_1 \cdot \cdots \cdot x_{i-1} \cdot x_{i+1} \cdot \cdots \cdot x_{n+1})$

- *If* $\beta(v) = x_i$ *for an* $i \in \{1, \dots, n\}$ *then* $val(v, a_1, \dots, a_n) =_{\text{def}} a_i$.
- *If* $\beta(v) \neq x_i$ *for all* $i \in \{1, \dots, n\}$ *then* $\beta(v) = f^m$ *for some Boolean function* $f^m \in B$ *and there exist* $v_1, \dots, v_m$ *with* $(v_1, v), \dots, (v_m, v) \in E$ *where* $\alpha((v_1, v)) < \dots < \alpha((v_m, v))$ *and we define*

$$val(v, a_1, \dots, a_n) =_{\text{def}} f^m(val(v_1, a_1, \dots, a_n), \dots, val(v_m, a_1, \dots, a_n))$$

Now, that we have defined B-circuits, we can make the connection to Boolean functions:

- *Every B-circuit C with n inputs defines an n-ary Boolean function* $f_C(a_1, \dots, a_n) =_{\text{def}} val(o, a_1, \dots, a_n)$ *for* $a_1, \dots, a_n \in \{0, 1\}$. *We say C* describes f_C. *By writing* C^n *we state that the arity of* f_C *is n. For* $a, a_1, \dots, a_n \in \{0, 1\}$ *we will often write* $C^n(a_1, \dots, a_n) = a$ *for* $f_C^n(a_1, \dots, a_n) = a$.
- *Let* B_1 *and* B_2 *be sets of Boolean functions and let* C_1 *be a* B_1*-circuit and* C_2 *be a* B_2*-circuit. We say* C_1 *is* equivalent *to* C_2 *and write* $C_1 \equiv C_2$ *if* f_{C_1} *can be derived from* f_{C_2} *by introducing 0 or more fictive variables or the other way round.*
- *Let* $\mathrm{C}(B)$ *be the set of all B-circuits.*
- *Let* $C = (V, E, \alpha, \beta, o)$ *be a B-circuit, then* $\mathrm{dual}(C) =_{\text{def}} (V, E, \alpha, \beta^*, o)$ *with*

$$\beta^*(v) =_{\text{def}} \begin{cases} \beta(v) & : \textit{if } \beta(v) \in \{x_1, \dots, x_n\} \\ \mathrm{dual}(\beta(v)) & : \textit{otherwise} \end{cases}$$

- *Let T be a set of B-circuits, then* $\mathrm{dual}(T) =_{\text{def}} \{\mathrm{dual}(C) | C \in T\}$.
- *Let C be a B-circuit. Then* $\mathrm{var}(C) =_{\text{def}} \mathrm{var}(f_C)$ *and* $\underline{\mathrm{var}}(C) =_{\text{def}} \underline{\mathrm{var}}(f_C)$.

We note that for a set B of Boolean functions and a B-circuit C holds that $\mathrm{dual}(C)$ is a $\mathrm{dual}(B)$-circuit, $f_{\mathrm{dual}(C)} = \mathrm{dual}(f_C)$ due to Proposition 2, and $\mathrm{C}(\mathrm{dual}(B)) = \mathrm{dual}(\mathrm{C}(B))$. Thus we are able to easily calculate for a Boolean function f, described by a B-circuit (where B is finite), its dual function, described as a $\mathrm{dual}(B)$-circuit.
Formally, we define *B-formulas* to be B-circuits with fan-out of at most 1. We will, however, use propositional formulas with the usual connectives $\vee$, $\wedge$, $\neg$, $\oplus$,... throughout the text to describe B-circuits and Boolean functions.
For some property P and a set of Boolean functions B , clearly the set $M =_{\text{def}} \{F | F$ is a B-formula with property $P\}$ is a subset of $N =_{\text{def}} \{C | C$ is a B-circuit with property $P\}$ and surely $M \leq_m^p N$ holds. So to show lower bounds of N it suffices to show them for M and to show upper bounds of M it suffices to show them for N. In this text we will make statements about the complexity of sets of Boolean circuits but these are correct for Boolean formulas as well. We will sometimes use formulas to show lower bounds of sets of circuits using the facts mentioned above.

2.2 Promises

In the segment that follows, A, B, A', and B' always denote sets, M denotes an algorithm and $\mathcal{K}$ denotes a complexity class.

Definition 4. M decides B with promise A *or* M decides the promise problem $B\wr_A$ *iff the following holds:*

- *If $x \in A \cap B$ then M accepts x*
- *If $x \in A \cap \overline{B}$ then M rejects x*

We will often call B the subject *of the promise problem. We also define a reflexive and transitive relation $\leq$: Write $B\wr_A \leq B'\wr_{A'}$ if there exists an $f \in \mathrm{FP}$ such that*

- *if $x \in A \cap B$ then $f(x) \in A' \cap B'$.*
- *if $x \in A \cap \overline{B}$ then $f(x) \in A' \cap \overline{B'}$.*

Write $B\wr_A \equiv B'\wr_{A'}$ if $B\wr_A \leq B'\wr_{A'}$ and $B'\wr_{A'} \leq B\wr_A$.

So if M has to decide $B\wr_A$, M is only required to work correctly on inputs from A; for all other inputs its behaviour is not defined. We will identify $B\wr_{\Sigma^*}$ with B. We write $B\wr_A \in \mathcal{K}$ if there is a $\mathcal{K}$-algorithm that decides $B\wr_A$. We say $B\wr_A$ is *hard* for $\mathcal{K}$ if $C \leq B\wr_A$ for every $C \in \mathcal{K}$ and $B\wr_A$ is *complete* for $\mathcal{K}$ if $B\wr_A \in \mathcal{K}$ and $B\wr_A$ is hard for $\mathcal{K}$.

Proposition 5.

1. *If $A' \subseteq A$ then $B\wr_{A'} \leq B\wr_A$.*
2. *$B\wr_A \equiv (A \cap B)\wr_A$.*

Lemma 6. *Let A, B and C be sets with $C \subseteq B$, $A \not\subseteq B$. If $C\wr_B \in \mathrm{P}$ then $C\wr_A \leq B\wr_A$.*

Proof. Since $C\wr_B \in \mathrm{P}$, there exists a $f \in \mathrm{P}$ such that

- $x \in B \cap C \Rightarrow f(x) = 1$
- $x \in B \cap \overline{C} \Rightarrow f(x) = 0$

Now we choose $x_0 \in A \cap \overline{B}$ and define

$$g(x) =_{\mathrm{def}} \begin{cases} x & : \text{ if } f(x) = 1 \\ x_0 & : \text{ if } f(x) = 0 \end{cases}$$

Obviously $g(x) \in \mathrm{FP}$.

- If $x \in A \cap C$ we get $x \in C = B \cap C$ and so $f(x) = 1$ and consequently $g(x) = x \in A \cap B$.
- If $x \in A \cap \overline{C}$:
  - **case 1:** If $x \in B$ then $x \in B \cap \overline{C}$ and so $f(x) = 0$ and we get $g(x) = x_0 \in A \cap \overline{B}$.
  - **case 2:** If $x \notin B$ then either $g(x) = x \in A \cap \overline{B}$ or $g(x) = x_0 \in A \cap \overline{B}$.

$\square$

3 Membership Problems of Post's Classes

Definition 7. *Let B be an arbitrary set of Boolean functions and let* $\mathrm{U} =_{\mathrm{def}} \{et, vel, non\}$*. Then*

$$\mathcal{M}(B) =_{\mathrm{def}} \{C | C \in \mathrm{C}(\mathrm{U}) \textit{ and } f_C \in B\}$$

is called MEMBER(B)-problem *for sets of Boolean functions.*

There are three reasons why we concentrate on U formulas. The first one is that we want to be able to represent every $f \in \mathrm{BF}$ with a $C \in \mathrm{C}(\mathrm{U})$, and for the second one, we want to be able to easily calculate the dual circuit of every $C \in \mathrm{C}(\mathrm{U})$; this dual circuit should also be a U-circuit. For that U has to be finite, complete and closed under duality such that we can utilize Proposition 2. So we could *not* have chosen e.g. $\{sh\}$ though it is complete, as this set is not closed under duality, but we could have chosen any other set that satisfies our demands. U simply is the one most readers will be familiar with.

Proposition 8. *Let B be a set of Boolean functions.*

1. $\mathrm{C}(B) \subseteq \mathcal{M}([B] \cup \{id\})$
2. $\mathcal{M}(\mathrm{dual}(B)) = \mathrm{dual}(\mathcal{M}(B))$

Definition 9. *For sets A and B of Boolean functions let*

$$\mathcal{M}(B\wr_A) =_{\mathrm{def}} \mathcal{M}(B)\wr_{\mathcal{M}(A)}$$

be called MEMBER(B) with promise A.

For every set B of Boolean functions we identify $\mathcal{M}(B\wr_{\mathrm{BF}})$ with $\mathcal{M}(B)$.
The subject of this thesis is to study for each closed class of Boolean functions, how difficult it is to decide, if we provide another closed class of Boolean functions as promise. Our goal will be to get a result for every possible combination. Now the problem arises that there are quite a number of possible combinations, so we need means that enable us to make statements over as many as possible combinations at once.

Proposition 10. *Let A, A', B, and B' be closed classes of Boolean functions and let $\mathcal{K}$ be a complexity class.*

1. *Let $\mathcal{M}(B\wr_A) \in \mathcal{K}$ and $\mathcal{M}(B'\wr_A) \in \mathcal{K}$. Then*
   - $\mathcal{M}((B_1 \cup B_2)\wr_A) \in \mathcal{K}$ *and*
   - $\mathcal{M}((B_1 \cap B_2)\wr_A) \in \mathcal{K}$

   holds.
2. $\mathcal{M}(B\wr_A) \equiv \mathcal{M}((B \cap A)\wr_A)$
3. $\mathcal{M}(B\wr_A) \equiv \mathcal{M}(\mathrm{dual}(B)\wr_{\mathrm{dual}(A)})$
4. *Let $A' \subseteq A$. Then* $\mathcal{M}(B\wr_{A'}) \leq \mathcal{M}(B\wr_A)$.
5. $\mathcal{M}(B\wr_A) \in co\mathrm{NP}$

This proposition is essential for this thesis, it will be referred to in nearly all theorems to come, for the points of this proposition help us finding strategies to handle a lot of subject/promise combinations. Point 2 for example, hints us to first have a look at promise problems with small subject classes such as I_2 since these classes are intersections of many of the closed classes. Point 3 enables us to concentrate on just the right (or the left) half of the inclusion graph (see Figure 1), as the dual closed class to a given one can be found, as if mirrored, on the other side of the line going through BF and $\emptyset$. So E is the dual closed class of V, S_0^2 is the dual one of S_2^1 and L is the dual one of itself, for example. Point 4 hints us to study preferably small promises when searching for lower bounds, because those bounds will surely hold for superset-promises, too. With point 5 as premise we will not have to care for the upper bound when we are trying to prove the *co*NP completeness of one of our promise sets.
We will now have a look at the complexity of Post's classes when not making promises about the input.

Theorem 11 (Wagner). *$\mathcal{M}(B)$ is co*NP*-complete for every closed class of Boolean functions B such that $B \notin \{\emptyset, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$. $\mathcal{M}(B) \in \mathrm{P}$ for $B \in \{\emptyset, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$.*

Proof. If B is from $\{\emptyset, \mathrm{BF}\}$, the claim is obvious, if it is from $\{\mathrm{R}_0, \mathrm{R}_1\, \mathrm{R}\}$ one only has to compute $C(0, \dots, 0)$ or $C(1, \dots, 1)$ or both to decide whether or not $C \in \mathcal{M}(B)$. Now, let TAUT$=_{\mathrm{def}}$ $\{C | C \in \mathrm{C(U)}$ and $C \equiv 1\}$, which is known to be *co*NP-complete.

- We reduce TAUT to $\mathcal{M}(B)$ where $B \in \{\mathrm{C}_0, \mathrm{C}\}$. For that, let C^n be a U-circuit and observe that $C(x_1, \dots, x_n\} \in$ TAUT if and only if $\overline{C(x_1, \dots, x_n)} \wedge z \in \mathcal{M}(B)$ for a $z \notin \mathrm{var}(C)$.

- Now let $B \in \mathrm{M} \cup \mathrm{L} \cup \mathrm{D} \cup \mathrm{S}_1^2) \setminus \mathrm{C}$ and let C be a U-circuit and $g(C) = (\overline{C(x_1, \dots, x_n)} \vee \overline{C(y_1, \dots, y_n)}) \wedge \overline{z} \vee u$ with $z, u \notin \mathrm{var}(C)$. We will show that $C \in$ TAUT if and only if $g(C) \in \mathcal{M}(B)$.
  - If $C \in$ TAUT then obviously $g(C) \in \mathcal{M}(\mathrm{I}_2) \subseteq \mathcal{M}(B)$.
  - If $C \notin$ TAUT then there is a tuple $\alpha = (a_1, \dots, a_n)$ where $a_i \in \{0, 1\}$ for $i \in \{1, \dots, n\}$, such that $C(\alpha) = 0$. We define $\overline{\alpha} =_{\mathrm{def}} (\overline{a_1}, \dots, \overline{a_n})$. Now we make the following observations:
    * $g(C)$ is not linear since both $g(C)(\alpha, \alpha, 0, 0) = 1 = g(C)(\alpha, \alpha, 0, 1)$ and $0 = g(C)(\alpha, \alpha, 1, 0) \neq g(C)(\alpha, \alpha, 1, 1) = 1$ hold.
    * $g(C)$ is not self-dual since $g(C)(\alpha, \alpha, 0, 0) \neq \overline{g(C)(\overline{\alpha}, \overline{\alpha}, 1, 1)}$.
    * $g(C)$ is not monotonic since although $(\alpha, \alpha, 0, 0) > (\alpha, \alpha, 1, 0)$ is true, so is $1 = g(C)(\alpha, \alpha, 0, 0) < g(C)(\alpha, \alpha, 1, 0) = 0$.
    * Finally $g(C) \notin \mathcal{M}(\mathrm{S}_1^2)$ since $g(C)(\alpha, 0^{n+2}) = 1 = g(C)(0^n, \alpha, 0, 0)$ holds.

  We still have to prove that $\mathcal{M}(B)$ is *co*NP complete for all B with $S_{02} \subseteq B \subseteq \mathrm{S}_0^2$. For this we can use the above and Proposition 10.3.

□

We start the discussion about the relative complexity of closed classes of Boolean functions with the easiest cases:

Theorem 12. *Let A and B be closed classes of Boolean functions.*

1. $\mathcal{M}(A\wr_A) \in \mathrm{L}$
2. *Let $B \in \{\emptyset, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$.*

3.1 Using Linear Classes and Subclasses of V and E as Promise

We have a look at promises that are closed classes appearing rather low in the hierarchy. For these we can expect our promise problems to tend to be easy, because the smaller a promise class gets, the more information an algorithm, that has to decide a promise problem, gets about its inputs.

We will see, that the information given by promises that are subclasses of V, E and L suffices to find the relevant variables of an input circuit. Knowing these, it is easy to compute equivalent circuits of a very regular form that can be used to find all the attributes we are looking for.

Definition 13. *Let B be a set of Boolean functions. We define*

$$\underline{\mathrm{VAR}} =_{\mathrm{def}} \{(C, x) | C \in \mathrm{C}(\mathrm{U}) \text{ and } x \in \underline{\mathrm{var}}(C)\}$$

and

$$\underline{\mathrm{VAR}}_B =_{\mathrm{def}} \underline{\mathrm{VAR}}\wr_{\{(C,x)|C \in \mathrm{C}(\mathrm{U}),\ f_C \in B\}}$$

Lemma 14. *Let A be a set of Boolean functions, such that $\underline{\mathrm{VAR}}_A \in \mathrm{P}$. Furthermore, fix a positive integer m and let B be a set of Boolean functions such that if $f \in B$ then $|\underline{\mathrm{var}}(f)| \leq m$. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$.*

Proof. Let $f_C^n \in A$. The first step is to compute the relevant variables of C and thereby count them. As $\underline{\mathrm{VAR}}_A \in \mathrm{P}$ this can be achieved in P. If $|\underline{\mathrm{var}}(C)| > m$ then $f_C^n \notin B$ and we reject. The next step is to calculate a partial truth table for f_C^n, using only the relevant variables. This truth table is of fixed size that is independent of the input length and can thus be calculated in polynomial time, since the value of a circuit can be calculated in polynomial time. The last step is to compare our truth table with the truth tables of all possible Boolean functions with at most m relevant variables and no fictive variables. The number of these tables is finite, so the last step can be calculated, too. $\square$

Lemma 15. *Let $X \in \{\mathrm{E}, \mathrm{V}, \mathrm{L}\}$. Then $\underline{\mathrm{VAR}}_X \in \mathrm{P}$.*

Proof. First note, that the value of a circuit C on a given input can always be computed in polynomial time.

- If C describes an $f^n \in \mathrm{E}$ we know, a formula $G =_{\mathrm{def}} c_0 \wedge (c_1 \vee x_1) \wedge (c_2 \vee x_2) \wedge \cdots \wedge (c_n \vee x_n)$ exists where c_i are constants for all $i \in \{0, \ldots, n\}$ and $C \equiv G$. Now it is easy to see that x_i is a relevant variable iff $C(1^{i-1}, 0, 1^{n-i}) \neq C(1^{i-1}, 1, 1^{n-i})$.
- If C describes an $f^n \in \mathrm{V}$, a formula $G =_{\mathrm{def}} c_0 \vee c_1x_1 \vee c_2x_2 \vee \cdots \vee c_nx_n$ exists with c_i constants for all $i \in \{0, \ldots, n\}$ and $C \equiv G$. Now it is easy to see that x_i is a relevant variable iff $C(0^{i-1}, 1, 0^{n-i}) \neq C(0^{i-1}, 0, 0^{n-i})$.

- If C describes an $f^n \in \mathrm{L}$, a formula $G =_{\mathrm{def}} c_0 \oplus c_1 x_1 \oplus c_2 x_2 \oplus \cdots \oplus c_n x_n$ exists with c_i constants for all $i \in \{0, \ldots, n\}$ and $C \equiv G$. Now it is easy to see that x_i is a relevant variable iff $C(0^{i-1}, 1, 0^{n-i}) \neq C(0^{i-1}, 0, 0^{n-i})$.

$\square$

Corollary 16. *Let B be a closed class of Boolean functions with $B \subseteq \mathrm{N}$ and A be a closed class of Boolean functions with $A \subseteq \mathrm{L}$ or $A \subseteq \mathrm{E}$ or $A \subseteq \mathrm{V}$. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$ holds.*

Proposition 17 ([JGK70]).

$$\mathrm{L}_3 = \{f | f \textit{ is linear Boolean function and } |\underline{\mathrm{var}}(f)| \equiv 1 \ (mod\ 2)\}.$$

Theorem 18. *Let A be a closed class of linear Boolean functions and B be an arbitrary closed class of Boolean functions. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$.*

Proof. Let $A = \mathrm{L}$. Due to Proposition 10.2 we only need to show the claim for sets B with $B \subseteq \mathrm{L}$. According to Corollary 16 the claim holds for $B \subseteq \mathrm{N}$ so we still have to show the claim for $B \in \{\mathrm{L}, \mathrm{L}_0, \mathrm{L}_1, \mathrm{L}_2, \mathrm{L}_3\}$.

1. The case $B = \mathrm{L}$ is trivial.
2. For $B \in \{\mathrm{L}_0, \mathrm{L}_1, \mathrm{L}_2\}$ observe that $\mathrm{L}_0 = \mathrm{L} \cap \mathrm{R}_0$, $\mathrm{L}_1 = \mathrm{L} \cap \mathrm{R}_1$ and $\mathrm{L}_2 = \mathrm{L} \cap \mathrm{R}$. So we can use Theorem 12.2 and Proposition 10.2.
3. For $B = \mathrm{L}_3$ we only have to determine whether the number of relevant variables of the input formula is odd, following Proposition 17. This is easy, as we have seen in Proposition 15.

For all $A \subset \mathrm{L}$ use 1. - 3., Corollary 16, and Proposition 10.4. $\square$

3.2 Using Monotonic Classes as Promise

We will begin our analysis of monotonic promises for low subject classes namely constant classes and classes of identities.

Lemma 19. *Let $A \subseteq \mathrm{M}$ and $B \subseteq \mathrm{I}$. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$ holds.*

Proof. We know that $\mathcal{M}(\mathrm{C}_0\wr_A) \in \mathrm{P}$ ($\mathcal{M}(\mathrm{C}_1\wr_A) \in \mathrm{P}$): For a given input circuit C that is known to describe a monotonic Boolean function, we only have to test whether or not $C(1, \ldots, 1) = 0$ ($C(0, \ldots, 0) = 1$) and $\mathcal{M}(\mathrm{C}\wr_A) \in \mathrm{P}$ holds since $\mathrm{C} = \mathrm{C}_0 \cup \mathrm{C}_1$ and Proposition 10.1. We also note, that $\mathrm{I}_0 = \mathrm{I}_2 \cup \mathrm{C}_0$, $\mathrm{I}_1 = \mathrm{I}_2 \cup \mathrm{C}_1$, and $\mathrm{I} = \mathrm{I}_0 \cup \mathrm{I}_1$; so, using Proposition 10.1 again, $\mathcal{M}(\mathrm{I}_2\wr_A) \in \mathrm{P}$ is still to prove. To do this we show the following claim: A monotonic Boolean function f^n is an identity function iff there is a unique m ($1 \leq m \leq n$) such that for $\alpha = (0^{m-1}10^{n-m})$ and $\beta = (1^{m-1}01^{n-m})$ holds that $f(\alpha) = 1$ and $f(\beta) = 0$:
If f is an identity function a unique m as described certainly exists. Observe, on the other hand, that for every $\gamma \in \{0,1\}^{m-1} \times \{1\} \times \{0,1\}^{n-m}$ holds $\alpha \leq \gamma$. Therefore, since f is monotonic $f(\gamma) = 1$. Furthermore for every $\delta \in \{0,1\}^{m-1} \times \{0\} \times \{0,1\}^{n-m}$, holds $\beta \geq \delta$ and therefore $f(\delta) = 0$, again due to the monotony of f. That means, that $f(a_1, \ldots, a_n) = a_m$ for all $a_i \in \{0,1\}$ such that $1 \leq i \leq n$; consequently f is the n-ary identity at position m. $\square$

Corollary 20. *$\mathcal{M}(B\wr_A) \in \mathrm{P}$ holds for every $A \subseteq \mathrm{M}$ and all B with $B \cap A \subseteq \mathrm{I}$.*

Theorem 21. *Let $A \subseteq \mathrm{M}$ be a closed class of Boolean functions and let $B \subseteq \mathrm{E}$ or $B \subseteq \mathrm{V}$ be another closed class of Boolean functions. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$.*

Proof. We first show $\mathrm{E}\wr_A \in \mathrm{P}$. For that let f_C be monotonic. If $C(1^{j-1}01^{n-j}) = 0$ then $C(x_1, \ldots, x_n) \equiv x_j \wedge C(x_1, \ldots, x_{j-1}, 1, x_{j+1}, \ldots, x_n)$. Let $J =_{\text{def}} \{j | C(1^{j-1}01^{n-j}) = 0\}$, let $C' =_{\text{def}} C(y_1, \ldots, y_n)$ where $y_i = x_i$ for all $i \notin J$, and $y_i = 1$ for all $i \in J$ and let $G =_{\text{def}} \bigwedge_{j \in J} x_j \wedge C'$. We observe that $G \equiv C$ and thus $f_C \in \mathrm{E}$ iff $f_{C'}$ is constant. To compute whether a monotonic f is from E_0 we have to ensure that $f \notin \mathrm{C}_1$ and $f \in \mathrm{E}$. Similarly to decide whether or not a monotonic f is from E_1 we have to ensure that $f \notin \mathrm{C}_0$ and $f \in \mathrm{E}$. To decide whether a monotonic f is from E_2 we have to ensure that it is no constant function and that it is from E. We have seen that all these tests are easy to perform and for the case $B \subseteq \mathrm{I}$, we can use Corollary 20. We still have to prove the claim for $B \subseteq \mathrm{V}$: For that we can use Proposition 10.3. □

Theorem 22. *Let A and B be closed classes of Boolean functions with $A \subseteq \mathrm{M}$ and $B \in \{\mathrm{M}, \mathrm{M}_0, \mathrm{M}_1, \mathrm{M}_2\}$. Then $\mathcal{M}(B\wr_A) \in \mathrm{P}$.*

Theorem 23. *$\mathcal{M}(\mathrm{S}_{10}\wr_{\mathrm{M}_2})$ is in P.*

Proof. For a Boolean function f^n we say f^n is *1-separating at position i* $(1 \leq i \leq n)$ if for all $a_1, \ldots, a_n \in \{0, 1\}$ with $f(a_1, \ldots, a_n) = 1$ holds: $a_i = 1$. We will prove the following observation:
If f is monotonic then f is 1-separating at position i if and only if $f(1^{i-1}01^{n-i}) = 0$.
Obviously, if $f(1^{i-1}01^{n-i}) = 1$ then f is not 1-separating at position i. On the other hand, if $f(1^{i-1}01^{n-i}) = 0$ then for all $\alpha = (a_1, \ldots, a_n) \in \{0, 1\}^n$ with $f(\alpha) = 1$ holds $a_i = 1$. To see this, let us assume that there is an α with $f(\alpha) = 1$ and $a_i = 0$. Then $\alpha \leq (1^{i-1}01^{n-i})$ and thus, because f is monotonic, $f(\alpha) \leq f(1^{i-1}01^{n-i}) = 0$, so $f(\alpha) = 0$. That is a contradiction, so f is 1-separating at position i. □

Definition 24. *Let B be a set of Boolean functions. Then*

$$\mathrm{EQ}^{\mathrm{F}}(B) =_{\text{def}} \{(G, H) |\ G \text{ and } H \text{ are } B\text{-Formulas and } G \equiv H\}$$

Theorem 25 ([Rei01]). *Let B be a set of Boolean functions with $\mathrm{S}_{10} \subseteq [B]$. Then $\mathrm{EQ}^{\mathrm{F}}(B)$ is coNP-complete.*

Theorem 26. *$\mathcal{M}(\mathrm{D}_2\wr_{\mathrm{M}_2})$ and $\mathcal{M}(S_{10}^2\wr_{M_2})$ are coNP-complete.*

Proof. We will reduce $\mathrm{EQ}^{\mathrm{F}}(\{et, vel\})$ to $\mathcal{M}(\mathrm{D}_2\wr_{\mathrm{M}_2})$ and $\mathcal{M}(S_{10}^2\wr_{\mathrm{M}_2})$. For this, let G and H be $\{et, vel\}$-Formulas and let

$$F(x_1, \ldots, x_n, y, z) =_{\text{def}} y \cdot G(x_1, \ldots, x_n) \vee z \cdot \overline{H(\overline{x_1}, \ldots, \overline{x_n})} \vee yz.$$

We are now able to make the following observations about F for arbitrary $a_1, \ldots, a_n \in \{0, 1\}$.

- $F \in \mathcal{M}(\mathrm{M}_2)$
- $F(a_1, \ldots, a_n, 1, 0) = G(a_1, \ldots, a_n)$

- $F(a_1,\dots,a_n,0,1) = \overline{H(\overline{a_1},\dots,\overline{a_n})}$
- $\overline{F(\overline{a_1},\dots,\overline{a_n},\overline{1},\overline{0})} = H(a_1,\dots,a_n)$
- $\overline{F(\overline{a_1},\dots,\overline{a_n},\overline{0},\overline{1})} = \overline{G(\overline{a_1},\dots,\overline{a_n})}$
- $F(a_1,\dots,a_n,0,0) = \overline{F(\overline{a_1},\dots,\overline{a_n},1,1)} = 0$
- $F(a_1,\dots,a_n,1,1) = \overline{F(\overline{a_1},\dots,\overline{a_n},0,0)} = 1$

So if $G \equiv H$ then $F(x_1,\dots,x_n,y,z) = \overline{F(\overline{x_1},\dots,\overline{x_n},\overline{y},\overline{z})}$ and therefore $F \in \mathcal{M}(\mathrm{D}_2)$. If, on the other hand, $G \not\equiv H$ then there are $a_1,\dots,a_n \in \{0,1\}$ with $H(a_1,\dots,a_n) = 0$ and $G(a_1,\dots,a_n) = 1$, without loss of generality. So $F(a_1,\dots,a_n,1,0) = G(a_1,\dots,a_n) = 1$ and $F(\overline{a_1},\dots,\overline{a_n},0,1) = \overline{H(a_1,\dots,a_n)} = 1$ and therefore F is not 1-separating of degree 2, i.e. $F \notin \mathcal{M}(S_{10}^2)$. $\square$

Theorem 27. *$\mathcal{M}(\mathrm{D}_2 \wr_{S_{10}^2})$ is co*NP*-complete.*

Proof. We will reduce $\mathcal{M}(\mathrm{D}_2 \wr_{M_2})$ to $\mathcal{M}(\mathrm{D}_2 \wr_{S_{10}^2})$. For an H from $\mathcal{M}(\mathrm{M}_2)$ let

$$F =_{\text{def}} H(x_1,\dots,x_n) \wedge \overline{H(\overline{x_1},\dots,\overline{x_n})}.$$

Obviously F is monotonic and because H is from R_2 it also holds that $F \in \mathcal{M}(R_1)$. We still need to show that F is 1-separating of degree 2. Let us suppose that $F \notin \mathcal{M}(S_1^2)$. Then there exist tuples $\alpha = (\alpha_1,\dots,\alpha_n), \beta = (\beta_1,\dots,\beta_n) \in \{0,1\}^n$ with $F(\alpha) = F(\beta) = 1$ and there is no $s \in \{1,\dots,n\}$ such that $\alpha_s = \beta_s = 1$. Therefore $\beta \leq \overline{\alpha}$ holds and because F is monotonic $F(\overline{\alpha}) = 1$. If $F(\alpha) = 1$ then $H(\alpha) \wedge \overline{H(\overline{\alpha})} = 1$ and consequently $\overline{H(\alpha)} = 1$. If at the same time $F(\overline{\alpha}) = 1$ then $H(\overline{\alpha}) \wedge \overline{H(\alpha)} = 1$ and consequently $\overline{H(\alpha)} = 1$ which is a contradiction, so F is 1-separating of degree 2.
Now let $H \in \mathrm{D}_2$. Then $H(\alpha) = \overline{H(\overline{\alpha})}$ holds for all $\alpha \in \{0,1\}^n$ so

$$\overline{F(\overline{\alpha})} = \overline{H(\overline{\alpha}) \wedge \overline{H(\alpha)}} = H(\alpha) \vee \overline{H(\overline{\alpha})} = H(\alpha) = F(\alpha)$$

holds and therefore $F \in \mathcal{M}(\mathrm{D}_2)$.
Let $H \notin \mathrm{D}_2$; then there is an $\alpha \in \{0,1\}^n$ with $H(\alpha) = H(\overline{\alpha})$ and we conclude

$$\overline{F(\overline{\alpha})} = \overline{H(\overline{\alpha}) \wedge \overline{F(\alpha)}} = F(\alpha) \vee \overline{F(\overline{\alpha})} = F(\alpha) \vee \overline{F(\alpha)} = 1$$

and

$$H(\alpha) = F(\alpha) \wedge \overline{F(\overline{\alpha})} = F(\alpha) \wedge \overline{F(\alpha)} = 0$$

so F is not self-dual. $\square$

At this point two questions remain: How difficult is $\mathcal{M}(S_{10}^n \wr_{\mathrm{M}_2})$ where $n > 2$ and how difficult is $\mathcal{M}(S_{10}^n \wr_{S_{10}^m})$ where $m > n$? All the remaining cases with monotonic promise can easily be derived from these.
These two cases have to remain open for the moment, however, we can state a relationship between the two.

Proposition 28. *Let C^n be a circuit that is 1-separating of degree p and let G^l be a circuit that is 1-separating of degree q. Then $C \wedge G$ is 1-separating of degree $max(p, q)$.*

Proof. To prove this, let without loss of generality $p \geq q$. Let furthermore $T \subseteq f^{-1}_{C\wedge G}(1)$ with $|T| = p$. Note that for all $\alpha = (a_1, \ldots, a_n, b_1, \ldots, b_l)$ with $a_1, \ldots, a_n, b_1, \ldots, b_l \in \{0, 1\}$ and $\alpha \in T$ both $C(a_1, \ldots, a_n) = 1$ and $G(b_1, \ldots, b_l) = 1$ have to hold, and therefore there is an s with $a_s = 1$ for all $\alpha \in T$; therefore T is 1-separating of level p and the claim is proven. $\square$

Lemma 29. $\mathcal{M}(\mathrm{S}^n_{10}\wr_{\mathrm{M}_2}) \leq \mathcal{M}(\mathrm{S}^n_{10}\wr_{\mathrm{S}^m_{10}})$ *if* $(1 < m < n)$.

Proof. Let $h_m = \bigvee_{i=1}^{m+1} y_1 \cdot y_2 \cdot \cdots \cdot y_{i-1} \cdot y_{i+1} \cdot \cdots \cdot y_{m+1}$ and $H^k(x_1, \ldots, x_k)$ be a formula. Now define $g(H) =_{\text{def}} H \wedge h_m$. If $H \in \mathrm{M}_2$ then $g(H) \in \mathrm{S}^m_{10}$ because of Proposition 28 and because H and h_m are monotonic.

- If $H^k \in \mathcal{M}(\mathrm{M}_2) \cap \mathcal{M}(\mathrm{S}^n_{10})$ then $g(H^k) \in \mathcal{M}(\mathrm{S}^m_{10}) \cap \mathcal{M}(\mathrm{S}^n_{10})$, again due to Proposition 28.
- If $H^k \in \mathcal{M}(\mathrm{M}_2) \cap \overline{\mathcal{M}(\mathrm{S}^n_{10})}$ then there are
$$\begin{aligned} \alpha_1 &= (a_{11}, \ldots, a_{1k}) \\ \alpha_2 &= (a_{21}, \ldots, a_{2k}) \\ &\vdots \\ \alpha_n &= (a_{n1}, \ldots, a_{nk}) \end{aligned}$$
with $\alpha_i \in \{0, 1\}^k$ and $H(\alpha_i) = 1$ and there is no $s \in \{1, \ldots, k\}$ with $a_{is} = 1$. for all $i \in \{1, \ldots, k\}$. Now define
$$\gamma_i =_{\text{def}} \begin{cases} (1^{i-1}01^{m-i+1}) & : \text{ if } i \leq k+1 \\ (1^{m+1}) & : \text{ otherwise} \end{cases}$$
Then $g(H)(\alpha_i, \gamma_i) = 1$ for all $i \in \{1, \ldots, n\}$ but neither the first k columns nor the last $m + 1$ columns consist solely of 1s and therefore $g(H) \in \mathcal{M}(\mathrm{S}^m_{10}) \cap \overline{\mathcal{M}(\mathrm{S}^n_{10})}$.

$\square$

That leaves $\mathcal{M}(S^n_{10}\wr_{\mathrm{M}_2})$ where $n > 2$ as unsolved problem. We have already seen that $\mathcal{M}(S^2_{10}\wr_{\mathrm{M}_2})$ is coNP-complete so one would suspect that the remaining case is coNP-complete as well. However, we have also seen that $\mathcal{M}(S_{10}\wr_{\mathrm{M}_2})$ is easy to decide.

3.3 Using Self-Dual Classes as Promise

Let us have a look at a truth table of the self-dual Boolean function f_C that is described by $C =_{\text{def}} x\overline{y} \vee x\overline{z} \vee \overline{yz}$ ($\{f_C\}$ is a base of D).

x	y	z	f_C
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

An attribute of every self-dual function can be observed in this truth table: The negated function values of the lower half of the table are mirror image like to the upper half. In fact a Boolean function is self-dual if and only if its truth table has this attribute. That implies that we can generate a self-dual Boolean function from an arbitrary Boolean function by introducing a new (non-fictive) variable and thus doubling the size of the truth table and adding in the new second half of the table function values that are negations of the mirror images of the first half, that contains the function values of the original function. We do exactly this in the following Lemma:

Lemma 30. *Let $C \in \mathcal{M}(\mathrm{BF})$ and let the function sd be defined as follows:*

$$sd(C) =_{\mathrm{def}} (\overline{C(\overline{x_1}, \ldots, \overline{x_n})} \wedge z) \vee (C(x_1, \ldots, x_n) \wedge \overline{z}), z \notin \mathrm{var}(C).$$

1. *$sd(C) \in \mathcal{M}(\mathrm{D})$*
2. *$\underline{\mathrm{var}}(C) \subseteq \underline{\mathrm{var}}(r(C))$*

Proof. 1. We write χ for $x_1, \ldots, x_n$ and $\overline{\chi}$ for $\overline{x_1}, \ldots, \overline{x_n}$.

$$\begin{aligned}
\overline{sd(C)(\overline{\chi}, \overline{z})} &= \neg((\overline{C(\chi)} \wedge \overline{z}) \vee (C(\overline{\chi}) \wedge z)) \\
&\equiv \neg(\overline{C(\chi)} \wedge \overline{z}) \wedge \neg(C(\overline{\chi}) \wedge z) \\
&\equiv (C(\chi) \vee z) \wedge (\overline{C(\overline{\chi})} \vee \overline{z}) \\
&\equiv (C(\chi) \wedge \overline{C(\overline{\chi})}) \vee \\
&\quad\ (\overline{C(\overline{\chi})} \wedge z) \vee (C(\chi) \wedge \overline{z}) \\
&\equiv (\overline{C(\overline{\chi})} \wedge z) \vee (C(\chi) \wedge \overline{z}) \\
&= sd(C)(\chi, z)
\end{aligned}$$

2. If $x_i \in \underline{\mathrm{var}}(C)$ then an $i \in \{1, \ldots, n\}$, $\alpha_0 \in \{0,1\}^{i-1}$ and $\alpha_1 \in \{0,1\}^{n-i}$ exist, such that $C(\alpha_0, 0, \alpha_1) \neq C(\alpha_0, 1, \alpha_1)$. Therefore $sd(C)(\alpha_0, 0, \alpha_1, 0) \neq sd(C)(\alpha_0, 1, \alpha_1, 0)$ holds. Consequently $x_i \in \underline{\mathrm{var}}(sd(C))$.

□

So deciding whether an n-ary Boolean function f is from a closed class B when providing the promise, that f is self-dual, is most likely as difficult as deciding whether an $n-1$-ary function g is in B without promise. We will in fact see that $\mathcal{M}(B) \equiv \mathcal{M}(B \wr_{\mathrm{D}})$ holds for all closed classes B that are not constant.

Theorem 31. $\mathcal{M}(\mathrm{I}_2 \wr_{\mathrm{D}_1})$ *is complete for co*NP.

Proof. We will reduce $\mathcal{M}(\mathrm{C}_0)$, which we know to be complete for *co*NP from Theorem 11, to $\mathcal{M}(\mathrm{I}_2 \wr_{\mathrm{D}_1})$. For an arbitrary k-ary circuit C define

$$v(C) =_{\mathrm{def}} C(x_1, \ldots, x_k) \vee C(\overline{y_1}, \ldots, \overline{y_k}) \text{ and}$$

$$r(C) =_{\mathrm{def}} (C(x_1, \ldots, x_k) \vee (x_1 \wedge \cdots \wedge x_k)) \wedge (x_1 \vee \cdots \vee x_k)$$

Furthermore let $sd(C)$ be defined as in Lemma 30. Note that for any circuit C^n and for $\alpha \in \{0,1\}^n \setminus \{(0,\ldots,0),(1,\ldots,1)\}$ holds $C^n(\alpha) = r(C^n)(\alpha)$, $r(C^n)(0,\ldots,0) = 0$ and $r(C^n)(1,\ldots,1) = 1$. So if $f_C \in \mathrm{R}_1$ holds, so does $f_C = f_{r(C)}$. In any case, if f_C is self-dual then $f_{r(C)}$ is self-dual, too.
Finally define $q(C) =_{\mathrm{def}} r \circ sd \circ v(C)$ which will be the function to reduce $\mathcal{M}(\mathrm{C}_0)$ to $\mathcal{M}(\mathrm{I}_2 \wr_{\mathrm{D}_1})$.
To see, that $q(C)$ is appropriate, let C be a circuit describing some function $f_C^n \in \mathrm{BF}$, with $n \geq 1$. Observe that $q(C) \in \mathcal{M}(\mathrm{D}_1)$ and have a look at the following argumentation:

- If $C \in \mathcal{M}(\mathrm{C}_0)$ then $v(C) \in \mathcal{M}(\mathrm{C}_0)$ and consequently $sd(v(C))(x_1,\ldots,x_n,y_1,\ldots,y_n,z) \equiv (1 \wedge z) \vee (0 \wedge \overline{z}) \equiv z$ and therefore $q(C)(x_1,\ldots,x_n,y_1,\ldots,y_n,z) \equiv (z \vee (z \wedge \bigwedge_{i=1}^n (x_i \wedge y_i))) \wedge (z \vee \bigvee_{i=1}^n (x_i \vee y_i)) \equiv z \wedge (z \vee \bigvee_{i=1}^n (x_i \vee y_i)) \equiv z$ and consequently $q(C) \in \mathcal{M}(\mathrm{D}_1) \cap \mathcal{M}(\mathrm{I}_2)$.
- Now let $C \notin \mathcal{M}(\mathrm{C}_0)$:

  case 1: If C describes a function from C_1, so does $v(C)$ and therefore $sd(v(C))(x_1,\ldots,x_n,y_1,\ldots,y_n,z) \equiv (0 \wedge z) \vee (1 \wedge \overline{z}) \equiv \overline{z}$. So $q(C)(b_1,\ldots,b_{2n},z) = \mathrm{non}(z)$ for all $(b_1,\ldots,b_{2n},a) \notin \{(0,\ldots,0),(1,\ldots,1)\}$, $q(C)(0,\ldots,0) = 0$ and $q(C)(1,\ldots,1) = 1$. Since C has at least one variable, $q(C)$ has at least three variables, and we get:
  $q(C)(1,0^{2n-1},0) = 1 \neq q(C)(0^{2n},0) = 0$ and
  $q(C)(0^n,1,0^{n-1},0) = 1 \neq q(C)(0^{2n},0) = 0$. So $|\underline{\mathrm{var}}(q(C))| > 1$ and that means that $q(C) \in \mathcal{M}(\mathrm{D}_1) \cap \overline{\mathcal{M}(\mathrm{I}_2)}$.

  case 2: $C \notin \mathcal{M}(\mathrm{C}_1)$: There exists an $i \in \{1 \ldots, n\}$, $\beta_0 \in \{0,1\}^{i-1}$ and $\beta_1 \in \{0,1\}^{n-i}$ such that $C(\beta_0,0,\beta_1) \neq C(\beta_0,1,\beta_1)$. Without loss of generality, let us assume that $C(\beta_0,0,\beta_1) = 0$. Then we get
  $v(C)(\beta_0,0,\beta_1,\overline{\beta_0},1,\overline{\beta_1}) \neq v(C)(\beta_0,1,\beta_1,\overline{\beta_0},1,\overline{\beta_1})$ and
  $v(C)(\beta_0,0,\beta_1,\overline{\beta_0},1,\overline{\beta_1}) \neq v(C)(\beta_0,0,\beta_1,\overline{\beta_0},0,\overline{\beta_1})$
  hence $|\underline{\mathrm{var}}(v(C))| > 1$. With Lemma 30 $|\underline{\mathrm{var}}(sd(v(C))| > 1$ holds. We already saw that $r(G)(\alpha) = G(\alpha)$ for all $\alpha \in \{0,1\}^n \setminus \{(0,\ldots,0),(1,\ldots,1)\}$ and every circuit G. In the above equations neither $(0,\ldots,0)$ nor $(1,\ldots,1)$ are used so $|\underline{\mathrm{var}}(q(C))| > 1$, and consequently $q(C) \in \mathcal{M}(\mathrm{D}_1) \cap \overline{\mathcal{M}(\mathrm{I}_2)}$.

□

Theorem 32. $\mathcal{M}(A \wr_{\mathrm{D}_2}) \in \mathrm{P}$ *for all closed classes* A *of Boolean functions.*

Proof. Observe, that either $\mathrm{D}_2 \subseteq A$ or $\mathrm{D}_2 \cap A \subseteq \mathrm{I}$. The first case is trivial, for the second, note that $\mathrm{D}_2 \subset \mathrm{M}$ and use Lemma 19 and Proposition 10.2. □

Corollary 33. *Let $A \in \{\mathrm{D}, \mathrm{D}_1\}$ and let $B \notin \{\emptyset, \mathrm{C}_0, \mathrm{C}_1, \mathrm{C}, \mathrm{D}_1, \mathrm{D}, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$ be closed classes of Boolean functions. Then $\mathcal{M}(B\wr_A)$ is complete for co*NP. *For all other closed classes B, $\mathcal{M}(B\wr_A) \in \mathrm{P}$ holds.*

Proof. If $B \in \{\emptyset, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$ we find an algorithm that decides $\mathcal{M}(B\wr_A)$ as we have seen in Theorem 12. For $B \in \{\mathrm{C}_0, \mathrm{C}_1, \mathrm{C}\}$ note that $A \cap B = \emptyset$, so with Proposition 10.2 and Theorem 12 these cases are easy, too. If $A = B = \mathrm{D}_1$ or if $A = B = \mathrm{D}$ we use Theorem 12.
For the case $\mathcal{M}(\mathrm{D}_1)\wr_\mathrm{D}$, we note that $\mathrm{D}_1 = \mathrm{D} \cap \mathrm{R}$ so we can use Proposition 10.2 again. Now consider those cases, where B is none of these classes: Due to Proposition 10.4 we can restrict the proof to the case $A = \mathrm{D}_1$. For all the remaining closed classes B note that $\mathrm{D}_1 \cap B \in \{\mathrm{I}_2, \mathrm{L}_2, \mathrm{D}_2\}$. Theorem 31 proved that $\mathrm{I}_2\wr_{\mathrm{D}_1}$ is complete for *co*NP. We know that $\mathcal{M}(\mathrm{I}_2\wr_{\mathrm{D}_2}) \in \mathrm{P}$ (Theorem 32) and $\mathrm{I}_2 \subseteq \mathrm{D}_2$. We also know that $\mathcal{M}(\mathrm{I}_2\wr_{\mathrm{L}_2}) \in \mathrm{P}$ (Theorem 18) and $\mathrm{I}_2 \subseteq \mathrm{L}_2$ holds. With Lemma 6 we get the completeness for *co*NP of $\mathcal{M}(\mathrm{D}_2\wr_{\mathrm{D}_1})$ and $\mathcal{M}(\mathrm{L}_2\wr_{\mathrm{D}_1})$ and with Proposition 10.2 our claim is proven. ☐

3.4 Using a-Separating Classes as Promise

In this chapter, if we talk of a, we always mean an a from $\{0, 1\}$. There is an infinite number of a-separating closed classes; for every $n \geq 2$ there is an a-separating closed class of level n. If A is a-separating of level k and B is a-separating of level m then $A \subseteq B$ if $k \leq m$ (see Figure 1). We will now show that providing the promise that a Boolean function f is a-separating of level k is most likely not sufficient to find a fast algorithm that decides whether or not f is a-separating of level $k + 1$.

Theorem 34. *$\mathcal{M}(\mathrm{S}_1^m\wr_{\mathrm{S}_1^k})$ and $\mathcal{M}(\mathrm{S}_1\wr_{\mathrm{S}_1^k})$ are complete for co*NP *where $2 \leq k < m$.*

Proof. We will reduce $\mathcal{M}(\mathrm{S}_1^m)$ to $\mathcal{M}(\mathrm{S}_1^m\wr_{\mathrm{S}_1^k})$ ($\mathcal{M}(\mathrm{S}_1)$ to $\mathcal{M}(\mathrm{S}_1\wr_{\mathrm{S}_1^k})$). To do this, let f_C^n be an arbitrary Boolean function described by the circuit C and let

$$G_k =_{\mathrm{def}} \bigvee_{i=1}^{k+1} x_1 \cdot \dots \cdot x_{i-1} \cdot x_{i+1} \cdot \dots \cdot x_{k+1} \text{ for all } k > 0.$$

Note that $f_{G_k} \in \mathrm{S}_1^k$ but $f_{G_k} \notin \mathrm{S}_1^{k+1}$ (see [JGK70], Table 1). Let furthermore $g(C) =_{\mathrm{def}} C \wedge G_k(y_1, \dots, y_{k+1})$ with $y_1, \dots, y_{k+1} \notin \mathrm{var}(C)$ be the reducing function. Observe that $g(C) \in \mathcal{M}(\mathrm{S}_1^k)$ due to Proposition 28.

- If $C \in \mathcal{M}(\mathrm{S}_1^m)$ ($C \in \mathcal{M}(\mathrm{S}_1)$) then, using Proposition 28 again, $g(C)$ is also 1-separating of level m (is also 1-separating) so obviously $g(C) \in \mathcal{M}(\mathrm{S}_1^k) \cap \mathcal{M}(S_1^m)$ ($g(C) \in \mathcal{M}(\mathrm{S}_1^k) \cap \mathcal{M}(\mathrm{S}_1)$)
- If $C \notin \mathcal{M}(\mathrm{S}_1^m)$ then there exist
$$\begin{aligned} \alpha_1 &= (a_{11}, \dots, a_{1n}) \\ \alpha_2 &= (a_{21}, \dots, a_{2n}) \\ &\vdots \\ \alpha_m &= (a_{m1}, \dots, a_{mn}),\ a_{ij} \in \{0, 1\} \end{aligned}$$
with $f_C^n(\alpha_i) = 1$ and there is no s ($1 \leq s \leq n$) with $a_{is} = 1$ for all $1 \leq i \leq m$. (If $f_C^n \notin \mathrm{S}_1$ then there exists an m such that $f_C^n \notin \mathrm{S}_1^m$ and the above statement holds.)

Now define

$$\gamma_i =_{\text{def}} \begin{cases} (1^{i-1}01^{k-i+1}) & : \text{ if } i \leq k+1 \\ (1^{k+1}) & : \text{ otherwise} \end{cases}$$

for $1 \leq i \leq m$. Now notice that the set $\{(\alpha_i, \gamma_i) | 1 \leq i \leq m\}$ is not 1-separating of level m, but $g(C)(\alpha_i, \gamma_i) = 1$ for all $1 \leq i \leq m$. Consequently $f_{g(C)}$ is not 1-separating of level m and $f_{g(C)}$ cannot be 1-separating; with other words: $g(C) \in \mathcal{M}(\mathrm{S}_1^k) \cap \overline{\mathcal{M}(\mathrm{S}_1^m)}$ and $g(C) \in \mathcal{M}(\mathrm{S}_1^k) \cap \overline{\mathcal{M}(\mathrm{S}_1)}$.

□

Corollary 35. *$\mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_{12}^k})$ and $\mathcal{M}(\mathrm{S}_{12} \wr_{\mathrm{S}_{12}^k})$ with $2 \leq k < m$ are complete for co*NP.

Proof. We will show $\mathcal{M}(\mathrm{S}_1^m \wr_{\mathrm{S}_1^k}) \leq \mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_{12}^k})$:
Let $f_C^n \in \mathrm{S}_1^k$ and $g(C) =_{\text{def}} C \vee (x_1 \cdot \dots \cdot x_n)$ where $\{x_1, \dots, x_n\} = \text{var}(C)$. Then $f_{g(C)}^n \in \mathrm{S}_{12}^k$ since $f_{g(C)}^n(1, \dots, 1) = 1$ and $f_{g(C)}^n$ is 1-separating of level k i.e. it might happen that $(1, \dots, 1) \notin f_C^{-1}(1)$ but $(1, \dots, 1) \in f_{g(C)}^{-1}(1)$, but this possible difference does not change the 1-separability of $f_{g(C)}^{-1}(1)$.

- If $f_C^n \in \mathrm{S}_1^m$ then $f_{g(C)}^n \in \mathrm{S}_{12}^m$.
- If $f_C^n \notin \mathrm{S}_1^m$ then α_i as in Lemma 34 exist and $f_{g(C)}^n(\alpha_i) = 1$ for all $1 \leq i \leq m$ and consequently $f_{g(C)}^n \notin \mathrm{S}_{12}^m$.

Analogously, $\mathcal{M}(\mathrm{S}_1 \wr_{\mathrm{S}_1^k}) \leq \mathcal{M}(S_{12} \wr_{\mathrm{S}_{12}^k})$ holds. □

Corollary 36. *Let $2 \leq k < m$.*

Then $\mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_1^k})$, $\mathcal{M}(\mathrm{S}_{12} \wr_{\mathrm{S}_1^k})$, $\mathcal{M}(\mathrm{S}_1 \wr_{\mathrm{S}_{12}^k})$, $\mathcal{M}(\mathrm{S}_1^m \wr_{\mathrm{S}_{12}^k})$,
$\mathcal{M}(\mathrm{S}_0^m \wr_{\mathrm{S}_0^k})$, $\mathcal{M}(\mathrm{S}_{02}^m \wr_{\mathrm{S}_{02}^k})$, $\mathcal{M}(\mathrm{S}_{02}^m \wr_{\mathrm{S}_0^k})$, $\mathcal{M}(\mathrm{S}_0^m \wr_{\mathrm{S}_{02}^k})$,
$\mathcal{M}(\mathrm{S}_0 \wr_{\mathrm{S}_0^k})$, $\mathcal{M}(\mathrm{S}_{02} \wr_{\mathrm{S}_{02}^k})$, $\mathcal{M}(\mathrm{S}_{02} \wr_{\mathrm{S}_0^k})$ *and* $\mathcal{M}(\mathrm{S}_{02} \wr_{\mathrm{S}_{02}^k})$

*are complete for co*NP.

Proof.

- As $\mathrm{S}_{12}^k \subset \mathrm{S}_1^k$, we can use Proposition 10.4 and Corollary 35 to get the result for $\mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_1^k})$ and $\mathcal{M}(\mathrm{S}_{12} \wr_{\mathrm{S}_1^k})$.
- With Proposition 10.2 we get $\mathcal{M}(\mathrm{S}_1^m \wr_{\mathrm{S}_{12}^k}) \equiv \mathcal{M}(\mathrm{S}_1^m \cap \mathrm{S}_{12}^k \wr_{\mathrm{S}_{12}^k}) = \mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_{12}^k})$ that we know to be *co*NP-complete due to Corollary 35.
- For *co*NP-hardness of the remaining sets see Proposition 10.3.

□

Theorem 37. $\mathcal{M}(\mathrm{S}_{12}^m \wr_{\mathrm{S}_1^m}) \in \mathrm{P}$, $\mathcal{M}(\mathrm{S}_{12} \wr_{\mathrm{S}_1}) \in \mathrm{P}$, $\mathcal{M}(\mathrm{S}_{02}^m \wr_{\mathrm{S}_0^m}) \in \mathrm{P}$ *and* $\mathcal{M}(\mathrm{S}_{02} \wr_{\mathrm{S}_0}) \in \mathrm{P}$.

Up until now we only dealt with a-separating promises and a-separating subject classes. To study the complexity of the remaining closed classes with a-separating promises we use a result from Lewis about the satisfiability of propositional formulas. It is widely known that the satisfiability problem for propositional formulas is NP complete. When we speak of propositional formulas in the context of satisfiability, we usually think of B-formulas with $[B] = \mathrm{BF}$. Lewis could show that for the satisfiability problem for B-formulas to be NP-complete it suffices that $\mathrm{S}_1 \subseteq [B]$. Of course, this result also holds for B-circuits, since B-formulas are B-circuits. Here is a formalized version of this result:

Definition 38. *For a set of Boolean functions B define*

$$\mathrm{SAT}(B) =_{\mathrm{def}} \{C^n | C^n \in \mathrm{C}(B) \text{ and there is an } \alpha \in \{0,1\}^n \text{ with } f_C^n(\alpha) = 1\}$$

Proposition 39 ([Lew79]). *Let B be a set of Boolean functions with $\mathrm{S}_1 \subseteq [B]$. Then* SAT*(B) is* NP*-complete.*

Theorem 40. *Let A be a closed class with $\mathrm{S}_1 \subseteq A$. Then $\mathcal{M}(\mathrm{C}\wr_A)$ and $\mathcal{M}(\mathrm{C}_0\wr_A)$ are complete for co*NP.

Proof. We define $g(x,y) =_{\mathrm{def}} et(x, non(y))$ and note that $[\{g\}] = \mathrm{S}_1$ (see Table 1). We will show that $\overline{\mathrm{SAT}(\{g\})} \leq \mathcal{M}(\mathrm{C}_0\wr_{\mathrm{S}_1})$ and $\overline{\mathrm{SAT}(\{g\})} \leq \mathcal{M}(\mathrm{C}\wr_{\mathrm{S}_1})$. For that, let the function f_C^n be in S_1. Now define $r(C) =_{\mathrm{def}} C \wedge z$, $z \notin \mathrm{var}(C)$. Note that $r(C) \in \mathcal{M}(\mathrm{S}_1)$, since $et(x,y) = g(x, g(x,y))$.

- If $C \in \overline{\mathrm{SAT}(\{g\})}$ then $r(C) \in \mathcal{M}(\mathrm{C}_0)$ and so $r(C) \in \mathcal{M}(\mathrm{S}_1) \cap \mathcal{M}(\mathrm{C}_0)$ and therefore $r(C) \in \mathcal{M}(\mathrm{S}_1) \cap \mathcal{M}(\mathrm{C})$, too.
- If $C \notin \overline{\mathrm{SAT}(\{g\})}$ then a tuple $\alpha \in \{0,1\}^n$ with $C(\alpha) = 1$ exists, so $r(C)(\alpha, 0) \neq r(C)(\alpha, 1)$ and it follows that $r(C) \in \mathcal{M}(\mathrm{S}_1) \cap \overline{\mathcal{M}(\mathrm{C})}$ and therefore $r(H) \in \mathcal{M}(\mathrm{S}_1) \cap \overline{\mathcal{M}(\mathrm{C}_0)}$, too.

For those remaining cases with $A \neq \mathrm{S}_1$ the claim can be proven using Proposition 10.4. ☐

Corollary 41. *Let A be a closed class with $\mathrm{S}_0 \subseteq A$. Then $\mathcal{M}(\mathrm{C}\wr_A)$ and $\mathcal{M}(\mathrm{C}_1\wr_A)$ are co*NP*-complete due to Theorem 40 and Proposition 10.3.*

Theorem 42. *Let A and B be closed classes of Boolean functions. Then $\mathcal{M}(B\wr_A)$ is co*NP*-complete if all the following statements hold:*

- $A \not\subseteq B$
- $\mathrm{S}_{12} \subseteq A$
- $B \notin \{\emptyset, \mathrm{C}_0, \mathrm{C}_1, \mathrm{C}, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$
- *For an integer $m \geq 2$ not both $A = \mathrm{S}_1^m$ and $B = \mathrm{S}_{12}^m$ hold.*

Proof. Let us first have a look at those cases where A is defined as described above and and for B additionally $\mathrm{S}_{12} \subseteq B$ holds: For these cases, Theorem 34 and Corollaries 35 and 36 already proved the claim. So we still have to show the remaining cases. For that let A and B be as described above and let additionally $\mathrm{S}_{12} \not\subseteq B$.

Then, with g being defined as in Theorem 40, $\overline{\mathrm{SAT}(\{g\})} \leq \mathcal{M}(B\wr_{\mathrm{S}_{12}})$ holds; this is sufficient to prove the whole claim because of Proposition 10.4. So take a circuit $C \in \mathrm{C}(\{g\})$ that describes a function $f_C^n \in \mathrm{S}_1$. Define

$$r(C) =_{\mathrm{def}} \overline{(\overline{C(x_1,\ldots,x_n)} \wedge C(y_1,\ldots,y_n) \wedge z_1)} \wedge z_2$$

and

$$s(C) =_{\mathrm{def}} r(C) \vee x_1 \cdot \cdots \cdot x_n \cdot y_1 \cdot \cdots \cdot y_n \cdot z_1 \cdot z_2.$$

Note that $r(C) \in \mathcal{M}(\mathrm{S}_1)$ since $g(x,y)$, $et(x,y)$ and f_C^n are in S_1. Also observe that $s(C)$ is from $\mathcal{M}(\mathrm{S}_{12})$ since $s(C)(1,\ldots,1) = 1$. Of course, since $s(C) \in \mathrm{S}_{12}$, it is 0-reproducing, too.

- If $C \in \overline{\mathrm{SAT}(\{g\})}$ then $f_{r(C)} = \mathrm{I}_{2n+2}^{2n+2} = f_{s(C)}$ and consequently $s(C) \in \mathcal{M}(\mathrm{S}_{12}) \cap \mathcal{M}(B)$.
- If $C \notin \overline{\mathrm{SAT}(\{g\})}$ then a tuple $\alpha \in \{0,1\}^n$ with $f_C^n(\alpha) = 1$ exists.
  - $0 = s(C)(0,\ldots,0,\alpha,0,0) \neq s(C)(0,\ldots,0,\alpha,0,1) = 1$ and $s(C)(0,\ldots,0,\alpha,1,0) = 0 = s(C)(0,\ldots,0,\alpha,1,1)$ so $s(C)$ is not linear.
  - $0 = s(C)(0,\ldots,0,\alpha,1,1) \neq \overline{s(C)}(1,\ldots,1,\overline{\alpha},0,0) = 1$ so $s(C)$ is not self-dual.
  - It holds, that $(0,\ldots,0,\alpha,1,1) > (0,\ldots,0,\alpha,0,1)$, but $0 = s(C)(0,\ldots,0,\alpha,1,1) < s(C)(0,\ldots,0,\alpha,0,1) = 1$ so $s(C)$ is not monotonic.
  - $s(C)(1,\ldots,1,\ 1,\ldots,1,\ 1,0) = s(C)(0,\ldots,0,\alpha,1,1) = 0$ so $s(C)$ is not 0-separating of level 2.

$\square$

Corollary 43. *Let A and B be closed classes of Boolean functions. Then $\mathcal{M}(B\wr_A)$ is co*NP*-complete if all the following statements hold:*

- $A \not\subseteq B$
- $\mathrm{S}_{02} \subseteq A$
- $B \notin \{\emptyset, \mathrm{C}_0, \mathrm{C}_1, \mathrm{C}, \mathrm{R}, \mathrm{R}_0, \mathrm{R}_1, \mathrm{BF}\}$
- *For an integer $m \geq 2$ not both $A = \mathrm{S}_0^m$ and $B = \mathrm{S}_{02}^m$ hold.*

4 Conclusion

Now to summarize the results of the last pages we can say that especially if we can guarantee a function to be in one of the lower closed classes we find fast algorithms for the $\mathcal{M}$-problem. This is a result that one will have expected since the smaller a promise is the more assumptions are made about the input of an algorithm, the more information such an algorithm has about the input to work with.
That we find fast algorithms for those cases where we have promises that are subsets of $\mathcal{M}(\mathrm{E})$ or $\mathcal{M}(\mathrm{V})$ or $\mathcal{M}(\mathrm{L})$ has its cause in the fact that formulas from such a set have

equivalent ones that are in a very regular form.
The case where $\mathcal{M}(\mathrm{D}_2)$ is the given promise is easy since we get a lot of information by stating that a circuit C is from this promise: C is self-dual, monotonic, 0- and 1-separating of degree 2, 0- and 1-reproducing!
As expected, the problems get harder when providing more general promises. So if e.g. $\mathcal{M}(\mathrm{D})$ is the promise the assumption about our input contains not much information; as we have seen, this information is worth one bit and thus not sufficient for a good algorithm. $\mathcal{M}(\mathrm{D}_1)$ isn't much better: We only get additional information about two input tuples and this information could easily have been calculated. This argument holds even more so for $\mathcal{M}(\mathrm{R})$, $\mathcal{M}(\mathrm{R}_0)$, and $\mathcal{M}(\mathrm{R}_1)$.
For promises that are not monotonic but contain the classes of per se 0- or 1-separating functions our $\mathcal{M}$-problems turned out to be hard almost everywhere, too, a result that was not visible from the start. Especially those cases where a 0- or 1-separating closed class of degree k as promise is not sufficient to gain a fast algorithm to decide 0- or 1-separating closed classes of degree $k+1$ are unexpected.
We finish this report without having solved the $\mathcal{M}$-problem for all possible $\mathcal{M}$-promises. For the monotonic closed classes above E or V there are still open problems, but solutions to all of these can easily be derived if we knew how difficult $\mathcal{M}(S_{10}^n \wr_{\mathrm{M}_2})$ is to decide.

5 Acknowledgments

I would like to thank Klaus Wagner for helping me a lot with this report; many of the best ideas used here came from him. Many thanks also to Steffen Reith for the introduction to the subject and for a lot of inspiring discussions.

References

[Coo71] S. A. Cook. The complexity of theorem proving procedures. *Proc. 3rd Ann. ACM Symp. on Theory of Computation*, pages 151–158, 1971.

[JGK70] S. W. Jablonski, G. P. Gawrilow, and W. B. Kudrajawzew. *Boolesche Funktionen und Postsche Klassen*. Akademie-Verlag, 1970.

[Lew79] Harry R. Lewis. Satisfiability problems for propositional calculi. *Mathematical Systems Theory*, 13:45–53, 1979.

[Pos41] E. L. Post. Two-value iterative systems, 1941.

[Rei01] Steffen Reith. *Generalized Satisfiability Problems*. PhD thesis, Julius-Maximilians-Universität Würzburg, 2001.

[RV00] S. Reith and H. Vollmer. Optimal satisfiability for propositional calculi and constraint satisfaction problems. In *Proceedings 25th International Symposium on Mathematical Foundations of Computer Science*, 2000.

[RW00] Steffen Reith and Klaus W. Wagner. The complexity of problems defined by boolean circuits. Technical Report 255, Institut für Informatik, Universität Würzburg, 2000.

[Vol99] H. Vollmer. *Introduction to Circuit Complexity*. Springer, 1999.

[Wag94] Klaus W. Wagner. *Einführung in die Theoretische Informatik*. Springer Verlag, 1994.

A Overview of the Results

$B\backslash A$	BF	R_0	R_1	R	M	M_0	M_2	S_1^2	S_1	S_{10}^2	S_{10}^{17}	D	D_1	D_2	L	E
BF	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢
R_0	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢
R_1	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢
R	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢
M	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
M_0	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
M_1	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
M_2	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
S_0^2	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🔴	🔴	🟢	🟢	🟢
S_0	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
S_1^2	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🟢
S_1	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🟢
S_{10}^2	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
S_{10}^{19}	🔴	🔴	🔴	🔴	🔵	🔵	🔵	🔴	🔴	🔵	🔵	🔴	🔴	🟢	🟢	🟢
S_{10}	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
D	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢
D_1	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢
D_2	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🔴	🔴	🟢	🟢	🟢
L	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
L_0	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
L_1	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
L_2	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
L_3	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
V	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
E	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
E_0	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
E_1	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
E_2	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
N	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
N_2	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
I	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
I_0	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
I_1	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
I_2	🔴	🔴	🔴	🔴	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🔴	🔴	🟢	🟢	🟢
C	🔴	🔴	🔴	🟢	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢	🟢
C_0	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🔴	🔴	🟢	🟢	🟢	🟢	🟢	🟢	🟢
C_1	🔴	🟢	🔴	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢
$\emptyset$	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢	🟢

🟢 $\mathcal{M}(B\wr_A)$ is in P 🔴 $\mathcal{M}(B\wr_A)$ is coNP complete 🔵 Difficulty of $\mathcal{M}(B\wr_A)$ not known yet