Journal of Universal Computer Science, vol. 2, no. 1 (1996), 533-548
submitted: 22/2/96, accepted: 12/4/96, appeared: 28/7/96 © Springer Pub. Co.

Ceilings of Monotone Boolean Functions

Paul E. Dunne
(University of Liverpool, U.K.
ped@csc.liv.ac.uk)

Abstract: This paper considers a particular relationship defined over pairs of n-argument monotone Boolean functions. The relationship is of interest since we can show that if (g, h) satisfy it then for any n-argument monotone Boolean function f there is a close relationship between the combinational and monotone network complexities of the function $(f \wedge g) \vee h$. We characterise the class of pairs of functions satisfying the relationship and show that it extends and encapsulates previous results concerning translations from combinational to monotone networks.

Key Words: Complexity measures, combinational networks, monotone Boolean functions,

Categories: F.1.1, F.1.3

1. Introduction

One of the most challenging problems facing researchers in computational complexity theory is to prove a superlinear lower bound on the Time Complexity of some decision problem in *NP*. The results of [Fischer and Pippenger 1979], [Schnorr 1976] show that for any decision problem f computable by a 2-tape (deterministic) Turing machine in time $T(n)$ for inputs of length n, there is a sequence of combinational networks, C_n, that compute f (restricted to inputs of length n) and having $O(T(n) \log T(n))$ gates. So, by virtue of these simulations, a superlinear lower bound on Time Complexity could be obtained by proving that the combinational network complexity of some explicitly defined family of Boolean functions was $\omega(n \log_2 n)$. Yet, despite the fact that 'almost all' n-argument Boolean functions require exponentially many 2-input gates to be used in their optimal combinational networks [Shannon 1949], the best lower bounds obtained to date for explicitly defined families of Boolean function are only linear. The lack of progress in analysing combinational network complexity has led to the investigation of restricted classes of Boolean network. The motivation underlying such investigations is twofold: first, in the hope that proof techniques for bounding the complexity of restricted models may yield insights into proof techniques for the most general form of combinational networks; secondly, to derive general network lower bounds by using efficient simulations of combinational networks by the restricted network class. The monotone network model — in which only 2-input $\wedge$ and $\vee$ gates are permitted — has been one of the most widely studied of these restricted models. Although this model is incomplete — networks within it can only realise the class of monotone Boolean functions — a number of important advances have been made within it. Undoubtedly the most significant of these are the methods for obtaining non-trivial bounds on specific monotone Boolean functions as described in [Alon and Boppana 1986], [Andreev 1985], and [Razborov 1985]. The techniques discovered have been of sufficient potency to permit the derivation of exponential lower bounds on monotone complexity. There remains, however, the problem of whether such results can be translated into similar bounds on combinational network size. The proof techniques developed for monotone Boolean networks rely heavily on combinatorial results associated with properties of monotone computation, and it seems unlikely that these techniques could be applied directly to combinational networks. Given that the monotone model is tractable with respect to proving non-trivial lower bound results, it is valuable to consider conditions under which lower bound results on monotone network complexity can be directly translated into corresponding results on combinational network size, i.e. to examine when efficient simulation of combinational by monotone networks exist. The use of *slice functions*, as advocated by [Berkowitz 1983], indicates that in many cases such translations exist. Nevertheless, although slice functions have a number of interesting properties, cf [Dunne 1986], [Dunne 1989], [Valiant 1986] it seems to be difficult to apply the available lower bound methods to them.

The purpose of the present paper is to examine the issue of combinational to monotone transformations by introducing a relationship between pairs of monotone Boolean functions and investigating its properties. This relationship offers a general method for translating from combinational to monotone complexity that subsumes the slice function transformation of [Berkowitz 1983]. In the next section of the paper we introduce some basic definitions and notations. In Section 3 the concept of a monotone function h being a *ceiling* of a monotone function g is introduced. The important property of this relationship is the following. If (g, h) is a pair of monotone Boolean functions such that h is a ceiling of g, then for any monotone Boolean function f, we can state a precise relationship between the combinational and monotone network complexities of the function $F \equiv f \wedge g \vee h$. Under suitable conditions, this is such that sufficiently large lower bounds on the *monotone* complexity of F imply similar lower bounds on the *combinational* complexity of f. A characterisation of those functions (g, h) such that h is a ceiling of g is given and some properties of this class of functions derived. In Section 4 we obtain some combinatorial estimates concerning the number of such functions within a specific class. In Section 5, we describe a mechanism for efficiently constructing a combinational network for f from networks which compute functions with related combinational and monotone complexity. Under suitable conditions this shows that if f does have large combinational complexity then a related function must have large monotone complexity. Conclusions are given in Section 6.

2. Preliminary Definitions and Notation

B_n denotes the set of n-input Boolean functions $f(\mathbf{X_n}) : \{0,1\}^n \rightarrow \{0,1\}$ with formal arguments $\mathbf{X_n} = < x_1, x_2, \ldots, x_n >$. M_n denotes the subset of B_n corresponding to the set of *monotone* Boolean functions, i.e. those functions with the property that $\forall\ 1 \leq i \leq n$

$$f(x_1, \ldots, x_{i-1}, 0, x_{i+1}, \ldots, x_n) \leq f(x_1, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)$$

(where the ordering $0 < 1$ is assumed).

If f, $g \in M_n$ with arguments $\mathbf{X_n}$ then $f \nleq g$ if $\exists\, \alpha \in \{0,1\}^n$ such that $f(\alpha) = 1$ and $g(\alpha) = 0$.

A *combinational network* is a directed acyclic graph comprising two types of node: input nodes which have in-degree 0 and are associated with elements of $\mathbf{X_n}$; and *gate* nodes which are associated with 2-input Boolean functions. A combinational network is considered to have a single *output* node, t say, which has outdegree 0. In the obvious way, an assignment $\alpha \in \{0,1\}^n$ to $\mathbf{X_n}$ induces a value $S(\alpha)$ at the output gate, t, of any combinational network S. S is said to compute $f \in B_n$ if $\forall\, \alpha \in \{0,1\}^n$ it holds that $f(\alpha) \equiv S(\alpha)$. A *monotone* Boolean network is defined in a similar manner to a combinational network except that the gate functions are restricted to be 2-input $\vee$ (disjunction) and 2-input $\wedge$ (conjunction). It is well known that $f \in M_n$ if and only if there is a monotone Boolean network, S, that computes f.

For $f \in B_n$, $\mathbf{C}(f)$ denotes the *combinational network complexity* of f, i.e. the number of gates in the smallest combinational network computing f. Similarly, for $f \in M_n$, $\mathbf{C^m}(f)$ denotes the number of gates in the smallest monotone network realising f, i.e. the *monotone network complexity* of f.

If $\mathbf{Y} \subseteq \mathbf{X_n}$ and $\pi \in \{0,1\}^{|\mathbf{Y}|}$ then $f^{|\,\mathbf{Y}:=\pi}$ denotes the *sub-function* of f (in $B_{n-|\mathbf{Y}|}$ and having arguments $\mathbf{X_n} - \mathbf{Y}$) obtained by fixing the variables in $\mathbf{Y}$ to π. For $\mathbf{Y} \subseteq \mathbf{X_n}$ a function of the form $\bigwedge_{x \in \mathbf{Y}} x$ is called a *product*; a function of the form $\bigvee_{x \in \mathbf{Y}} x$ is called a *sum*. It is well known that any $f \in M_n$ has unique minimal representations as a disjunction of product terms (DNF) and as a conjunction of sum terms (CNF). The set of product terms (sum terms) occurring in such representations of f are called the *prime implicants of f* (*prime clauses of f*) these sets being denoted $\mathbf{PI}(f)$ ($\mathbf{PC}(f)$). For a product p or sum q, $|\,p\,|$ (resp. $|\,q\,|$) will denote the number of variables defining p (resp. q).

For further background on Boolean complexity theory the reader is referred to [Dunne 1988].

3. Ceilings of Monotone Functions

Definition 3.1: Let $g, h \in M_n$ with formal arguments $\mathbf{X_n}$. h is a *ceiling* of g if

$$\forall\ x_i \in \mathbf{X_n} \qquad g^{|\,x_i:=0} \;\leq\; h^{|\,x_i:=1} \qquad \bullet$$

The reason why this relationship is of interest is the following result.

Theorem 3.1: *If (g, h) are such that h is a ceiling of g then for any $f \in M_n$ with arguments $\mathbf{X_n}$ it holds*

$$\mathbf{C^m}((f \wedge g) \vee h) \;\leq\; 6\,\mathbf{C}((f \wedge g) \vee h) + n\,\mathbf{C^m}(h)$$

Proof: Let $F(\mathbf{X_n}) \equiv ((f \wedge g) \vee h)(\mathbf{X_n})$. It is well known, e.g. [Dunne 1988] (pp.239-241), that any optimal combinational network, T, for F can be transformed into a network in which negation is only applied to input nodes and the only gate operations used are $\wedge$ and $\vee$. This network contains at most $6\mathbf{C}(F)$ gates (not counting the negations on input nodes). Thus if each instance $\neg x_i$ can be replaced by some monotone Boolean function $h_i \in M_{n-1}$ then we can construct a *monotone* network that still computes $F(\mathbf{X_n})$. In [Dunne 1984] it is proved that h_i is a suitable monotone Boolean function with which to replace $\neg x_i$ if and only if h_i satisfies

$$(F^{|\,x_i:=0})(\mathbf{X_n} - \{x_i\}) \;\leq\; h_i(\mathbf{X_n} - \{x_i\}) \;\leq\; (F^{|\,x_i:=1})(\mathbf{X_n} - \{x_i\})$$

To prove the theorem it suffices to show that choosing $h_i = h^{|\,x_i:=1}$ yields a correct replacement. We have

$$\begin{aligned}
F^{|x_i:=0} &\equiv (f \wedge g)^{|x_i:=0} \vee h^{|x_i:=0} \\
&\leq g^{|x_i:=0} \vee h^{|x_i:=0} \\
&\leq g^{|x_i:=0} \vee h^{|x_i:=1} && (3.1)\\
&\equiv h^{|x_i:=1} && (3.2)
\end{aligned}$$

$$\leq \ (f \wedge g)^{|x_i:=1} \ \vee \ h^{|x_i:=1}$$
$$\equiv \ F^{|x_i:=1}$$

as required. (3.1) follows from the monotonicity of h; (3.2) from the definition of ceiling and the fact that for monotone Boolean functions v and w it holds $(v \leq w) \Rightarrow (v \vee w \equiv w)$. □.

Definition 3.2: If h is a ceiling of g and $f \in M_n$, then the (g,h)*-variant* of f, is the function $f \wedge g \vee h$. •

We note, in passing, that (T^n_k, T^n_{k+1}) (where T^n_m is the m-th *threshold function*, i.e. the function which is 1 if and only if at least m inputs are 1) has the property that T^n_{k+1} is a ceiling of T^n_k and these give rise to the slice function transformation of Berkowitz. Similarly one extension of slice functions — the construction $(g, \bigvee_{i=1}^{n} (x_i \wedge g^{|x_i:=0}))$ — introduced in [Dunne 1992] gives a mechanism for constructing a ceiling of any function g.

The set of (g, h) of functions in M_n such that h is a ceiling of g is characterised by a relationship between the prime clauses (or prime implicants) of the functions.

Definition 3.3: For $f \in M_n$ with arguments $\mathbf{X_n}$,

$$\vee\text{–}core\,(f) \ = \ \{\, q : q \textit{ is a sum and } \forall\, x \in \mathbf{X_n} \textit{ with } x \nleq q \textit{ it holds that } f \leq x \vee q \,\}$$

$$\wedge\text{–}core\,(f) \ = \ \{\, p : p \textit{ is a product and } \forall\, x \in \mathbf{X_n} \textit{ with } p \nleq x \textit{ it holds that } x \wedge p \leq f \}$$

If Q is a set of sums,

$$\wedge\text{–}cover\,(Q) \ = \ \bigcup_{Q' \subseteq Q} \{\, \bigwedge_{q \in Q'} q \,\}$$

Similarly if P is a set of products

$$\vee\text{–}cover\,(P) \ = \ \bigcup_{P' \subseteq P} \{\, \bigvee_{p \in P'} p \,\} \qquad \bullet$$

For example, suppose that $f \in M_3$ is the function

$$f(x_1, x_2, x_3) \ = \ (x_1 \vee x_2) \wedge (x_1 \vee x_3)$$
$$\equiv \ x_1 \vee (x_2 \wedge x_3).$$

For this choice of f, Definition 3.3, gives

$$\vee\text{–}core(f) \ = \ \{\, x_1 \vee x_2 \vee x_3 \ ; \ x_1 \vee x_2 \ ; \ x_1 \vee x_3 \ ; \ x_2 \vee x_3 \ ; \ x_1 \,\}$$

$$\wedge\text{–}core(f) \ = \ \{\, x_1 \wedge x_2 \wedge x_3 \ ; \ x_1 \wedge x_2 \ ; \ x_1 \wedge x_3 \ ; \ x_2 \wedge x_3 \ ; \ x_1 \ ; \ x_2 \ ; \ x_3 \,\}$$

Theorem 3.2: $\forall\, g \in M_n$, $\forall\, h \in M_n$ *with g and h having formal arguments* $\mathbf{X_n}$

$$h \textit{ is a ceiling of } g \quad \Leftrightarrow \quad h \in \wedge\text{–}cover\,(\vee\text{–}core\,(g))$$
$$\Leftrightarrow \quad g \in \vee\text{–}cover\,(\wedge\text{–}core\,(h))$$

Proof: It suffices to prove the first equivalence since a dual argument then establishes the second.

$\Leftarrow$: Suppose $h \in \wedge\text{–}cover\,(\vee\text{–}core\,(g))$. We prove that h is a ceiling of g in two stages. We first establish that for any $q \in \vee\text{–}core(g)$, q is a ceiling of g. It is then shown that for any set of sums, Q, with the property that $\forall\, q \in Q$, q is a ceiling of g, it holds that $\forall\, h \in \wedge\text{–}cover\,(Q)$, h is a ceiling of g. Clearly these two results yield the $\Leftarrow$ implication.

Let $q \in \vee\text{–}core\,(g)$. Consider any $x_i \in \mathbf{X_n}$. If $x_i \leq q$ then $g^{\,|x_i:=0} \leq q^{\,|x_i:=1}$ since $q^{\,|x_i:=1} = 1$. On the other hand if $x_i \nleq q$ then from the definition of $\vee\text{–}core(g)$ we have that $g \leq x_i \vee q$. Thus

$$g^{\,|x_i:=0} \leq (x_i \vee q)^{|x_i:=0} = q \leq q^{\,|x_i:=1}$$

It follows that $\forall x_i \in \mathbf{X_n}$, $g^{\,|x_i:=0} \leq q^{\,|x_i:=1}$, i.e. that q is a ceiling of g.

Now suppose Q is a set of products such that $\forall q \in Q$, q is a ceiling of g. Let $h \in \wedge\text{–}cover\,(Q)$, so that $h \equiv \bigwedge_{q \in Q'} q$, for some subset Q' of Q. Since q is a ceiling of g for every $q \in Q'$ we have

$$\forall\, x_i \in \mathbf{X_n} \quad g^{\,|x_i:=0} \;\leq\; q^{\,|x_i:=1}$$

Hence

$$\forall x_i \in \mathbf{X_n} \quad g^{\,|x_i:=0} \equiv \bigwedge_{q \in Q'} g^{\,|x_i:=0} \leq \bigwedge_{q \in Q'} q^{\,|x_i:=1} \;\equiv\; h^{\,|x_i:=1}$$

proving that h is a ceiling of g.

$\Rightarrow$: Suppose that h is a ceiling of g. Let $h = \bigwedge_{q \in Q} q$, for some set of sums Q. It is sufficient to show that $\forall q \in Q$, $q \in \vee\text{–}core(g)$.

Let $q \in Q$, so that $h \leq q$. Since h is a ceiling of g, it follows that $\forall x_i \in \mathbf{X_n}$

$$g^{\,|x_i:=0} \leq h^{\,|x_i:=1} \leq q^{\,|x_i:=1}$$

Hence, $\forall x_i \in \mathbf{X_n}$, $g^{\,|x_i:=0} \equiv q^{\,|x_i:=1} \wedge g^{\,|x_i:=0}$.

But $\forall x_i \in \mathbf{X_n}$

$$\begin{aligned} g &\equiv (x_i \vee g^{\,|x_i:=0}) \wedge g^{\,|x_i:=1} \\ &\equiv (x_i \vee (q^{\,|x_i:=1} \wedge g^{\,|x_i:=0})) \wedge g^{\,|x_i:=1} \\ &\equiv (x_i \vee q^{\,|x_i:=1}) \wedge (x_i \vee g^{\,|x_i:=0}) \wedge g^{\,|x_i:=1} \\ &\leq (x_i \vee q^{\,|x_i:=1}) \end{aligned}$$

Now if $x_i \nleq q$, then $q^{\,|x_i:=1} \equiv q$ and thus $\forall x \in \mathbf{X_n}$ such that $x \nleq q$ we have

$$g \;\leq\; (x \vee q^{\,|x:=1}) \;\equiv\; x \vee q$$

proving that $q \in \vee\text{–}core(g)$ as required. $\square$.

For the example $f(x_1, x_2, x_3) = x_1 \vee (x_2 \wedge x_3)$, from Theorem 3.2, using the definitions of $\wedge$-cover and $\vee$-cover, the table below (in which we write xy for $x \wedge y$) enumerates all of the appropriate g and h for f.

$\{h : h$ *is a ceiling of* $f\}$	$\{g : f$ *is a ceiling of* $g\}$
1	0
$x_1 \vee x_2 \vee x_3$	$x_1 x_2 x_3$
$x_1 \vee x_2,\; x_1 \vee x_3,\; x_2 \vee x_3$	$x_1 x_2,\; x_1 x_3,\; x_2 x_3$
x_1	$x_1,\; x_2,\; x_3$
$x_1(x_2 \vee x_3)$	$x_1(x_2 \vee x_3),\; x_2(x_1 \vee x_3),\; x_3(x_1 \vee x_2)$
$x_1 \vee x_2 x_3,\; x_2 \vee x_1 x_3,\; x_3 \vee x_1 x_2$	$x_1 \vee x_2 x_3,\; x_2 \vee x_1 x_3,\; x_3 \vee x_1 x_2$
$(x_1 \vee x_2)(x_1 \vee x_3)(x_2 \vee x_3)$	$x_1 x_2 \vee x_1 x_3 \vee x_2 x_3$

Table 1: Ceiling Properties of $f = x_1 \vee (x_2 \wedge x_3)$

The next result shows that the construction $f \wedge g \vee h$, where h is a ceiling of g is, in a weak sense, equivalent to the characterisation of functions which can replace $\neg x_i$ as described in the proof of Theorem 3.1.

Theorem 3.3: *Let $f \in M_n$ with arguments $\mathbf{X_n}$.*

i. *If $f(\mathbf{X_n}) \equiv (F \wedge G \vee H)(\mathbf{X_n})$ where $H(\mathbf{X_n})$ is a ceiling of $G(\mathbf{X_n})$ and $F, G, H \in M_n$ then $\forall\, x_i \in \mathbf{X_n}$, $\exists\, h_i \in M_{n-1}$ (defined only in terms of H) such that $f^{|\, x_i := 0} \leq h_i \leq f^{|\, x_i := 1}$.*

ii. *If $\forall\, x_i \in \mathbf{X_n}$, $\exists\, h_i \in M_{n-1}$ such that $f^{|\, x_i := 0} \leq h_i \leq f^{|\, x_i := 1}$ then $\exists\, G, H \in M_n$ (defined only in terms of h_i) such that $f \equiv F \wedge G \vee H$ (for some $F \in M_n$) and with H a ceiling of G*

Proof: (i) is immediate from the definition of ceiling, by choosing $h_i = H^{|\, x_i := 1}$. For (ii), since $\forall\, x_i \in \mathbf{X_n}$

$$f^{|\, x_i := 0} \;\equiv\; h_i \;\wedge\; f^{|\, x_i := 0} \quad ; \quad f^{|\, x_i := 1} \;\equiv\; h_i \;\vee\; f^{|\, x_i := 1}$$

and

$$f \;\equiv\; f^{|\, x_i := 0} \vee x_i \wedge f^{|\, x_i := 1} \;\equiv\; (x_i \vee f^{|\, x_i := 0}) \wedge f^{|\, x_i := 1}$$

we have,

$$f \;\equiv\; f \wedge (x_i \vee h_i) \quad ; \quad f \;\equiv\; x_i \wedge h_i \vee f$$

Hence

$$f \;\equiv\; f \wedge \bigwedge_{i=1}^{n} (x_i \vee h_i) \;\vee\; \bigvee_{i=1}^{n} (x_i \wedge h_i)$$

Let $G(\mathbf{X_n}) \equiv \bigwedge_{i=1}^{n} (x_i \vee h_i)$, $H(\mathbf{X_n}) \equiv \bigvee_{i=1}^{n} (x_i \wedge h_i)$. Then $G^{|\, x_i := 0} \leq h_i \leq H^{|\, x_i := 1}$ and thus H is a ceiling of G. □

An immediate consequence of Theorem 3.2 is that we can define functionals:

$$Min : M_n \rightarrow M_n \quad ; \quad Max : M_n \rightarrow M_n$$

by

$$Min(g)(\mathbf{X_n}) \;=\; \Big(\bigwedge_{q \,\in \vee - core(g)} q\Big)(\mathbf{X_n})$$

$$Max(h)(\mathbf{X_n}) \;=\; \Big(\bigvee_{p \,\in \wedge - core(h)} p\Big)(\mathbf{X_n})$$

and for these, it is easy to see that

- $\forall\, h \in M_n$ $Min(g) \leq h$ if and only if h is a ceiling of g.
- $\forall\, g \in M_n$ $g \leq Max(h)$ if and only if h is a ceiling of g.

With our example function, $f = x_1 \vee (x_2 \wedge x_3)$, we get that

$$Min(f) = x_1 \wedge (x_2 \vee x_3) \quad ; \quad Max(f) = x_1 \vee x_2 \vee x_3.$$

It may be the case, however, that $Min(f)$ and $Max(f)$ are uninteresting functions compared with f. This behaviour is illustrated by

Lemma 3.1: *$\forall\, f \in M_n$ with arguments $\mathbf{X_n}$ it holds that*

$$Min(f) \;\leq\; f \;\leq\; Max(f)$$

Proof: Consider any prime clause, q say, of f. Clearly since $\forall\, x_i \in \mathbf{X_n}$ it is the case that $f \leq q \leq q \vee x_i$, it follows that $q \in \vee{-}core(f)$. Similarly, for any prime implicant, p, of f we have $p \in \wedge{-}core(f)$. It follows that

$$Min(f) \;\equiv\; f \wedge Min(f) \quad ; \quad Max(f) \;\equiv\; f \vee Max(f)$$

from which the lemma is immediate. □

If it is the case that $g = Min(g)$, then for any function h such that h is a ceiling of g and any $f \in M_n$ it holds that

$$(f \wedge g) \vee h \equiv (f \wedge g) \vee Min(g) \vee h \equiv Min(g) \vee h \equiv h$$

Similarly if $h = Max(h)$ then

$$(f \wedge g) \vee h \equiv h$$

Thus in such cases the translation described by Theorem 3.1 would be of no interest. We therefore wish to identify necessary and sufficient conditions on $f \in M_n$ which will establish when $Min(f) < f$ and similarly $f < Max(f)$.

Definition 3.4: Let $f \in M_n$ with arguments $\mathbf{X_n}$. We say that f has a *disjunctive soft-core* (alternatively f is a *disjunctive soft-core function*) if

$$\forall\ q \in \vee\text{–}core(f) \quad f \le q$$

f has a *conjunctive soft-core* (alternatively, f is a *conjunctive soft-core function*) if

$$\forall\ p \in \wedge\text{–}core(f) \quad p \le f$$

f has a *disjunctive hard-core* (resp. *conjunctive hard-core*) if $\exists\, q \in \vee\text{–}core(f)$ (resp. $\exists\, p \in \wedge\text{–}core(f)$) such that $f \nleq q$ (resp. $p \nleq f$). •

Theorem 3.4: $\forall f \in M_n$ *with arguments* $\mathbf{X_n}$:

i. $Min(f) < f$ *if and only if f has a disjunctive hard-core.*

ii. $f < Max(f)$ *if and only if f has a conjunctive hard-core.*

Proof: We prove (i) only, since (ii) follows by a dual argument. Let $f \in M_n$ with arguments $\mathbf{X_n}$. Suppose that $Min(f) < f$. By definition $Min(f) = \bigwedge_{q \in \vee\text{–}core(f)} q$ and, since $\mathbf{PC}(f) \subseteq \vee\text{–}core(f)$, it follows that

$$Min(f) \equiv f \wedge \left(\bigwedge_{q \in Q} q \right)$$

where

$$Q = \{\ q : f \nleq q \ \ and\ \ q \in \vee\text{–}core(f)\ \}$$

If $Q = \emptyset$ then $Min(f) = f$. Since $Min(f) < f$ so Q is non-empty, i.e. f has a disjunctive hard-core.

On the other hand, suppose f has a disjunctive hard-core, Q say, and let $q \in Q$. By definition $f \nleq q$ so there exists $\pi \in \{0,1\}^n$ such that $f^{|\mathbf{X_n} := \pi} = 1$ and $q^{|\mathbf{X_n} := \pi} = 0$. $q \in \vee\text{–}core(f)$ and therefore $Min(f) \equiv q \wedge Min(f)$ thus $Min(f)^{|\mathbf{X_n} := \pi} = 0$ also. This proves that $Min(f) < f$ as claimed. □

4. The number of hard-core functions

In this section we give some estimates on the number of monotone Boolean functions within a certain class that have disjunctive hard-cores. In particular we concentrate on the class consisting of the monotone duals of $(k+1)$-homogeneous functions, denoted $\tilde{Q}_{n,k+1}$. Thus

$$\tilde{Q}_{n,k+1} =_{def} \{\, f \in M_n : \forall\, q \in \mathbf{PC}(f) \quad |\, q\, | = k+1 \,\}$$

Let $P_r(n)$ denote the set of all subsets of size r drawn from $\{1, 2, \ldots, n\}$ and for a set S, let 2^S denote the set of all subsets of S. Since any subset $S \subseteq P_{k+1}(n)$ uniquely describes the variables defining the prime clauses of some

$f \in \tilde{Q}_{n,k+1}$ it follows that

$$| \tilde{Q}_{n,k+1} | = 2^{\binom{n}{k+1}}$$

Define the functional $\chi : 2^{P_{k+1}(n)} \to \tilde{Q}_{n,k+1}$ by

$$\chi(\{ s_1, s_2, \ldots, s_r \}) = \bigwedge_{i=1}^{r} (\bigvee_{j \in s_i} x_j)$$

In this way if $C_{n,k+1} \subseteq \tilde{Q}_{n,k+1}$ is some property of functions in $\tilde{Q}_{n,k+1}$ the problem of determining the number of functions in $C_{n,k+1}$ is equivalent to counting the number of subsets S of $P_{k+1}(n)$ such that $\chi(S) \in C_{n,k+1}$.

The specific property $C_{n,k+1}$ that we are interested in is that of a function having a disjunctive hard-core. The definition below introduces the terminology we shall use to describe the corresponding subsets of $P_{k+1}(n)$.

Definition 4.1: Let $S \subseteq P_{k+1}(n)$. S has a *hard-core* (equivalently S is a *hard-core set*) if $\exists \{ s_1, s_2, \ldots, s_{n-k} \} \subseteq S$ such that $| \bigcap_{i=1}^{n-k} s_i | = k$. For $t \in P_k(n)$, we say that a set $S \subseteq P_{k+1}(n)$ *covers* the *hard-core element* t if $\exists \{ s_1, s_2, \ldots, s_{n-k} \} \subseteq S$ such that $\bigcap_{i=1}^{n-k} s_i = t$. It should be noted that $S \subseteq P_{k+1}(n)$ may cover several different hard-core elements. Finally we introduce the following sets:

$HC(n, k, r) =_{def} \{ S \subseteq P_{k+1}(n) : |S| = r \textit{ and } S \textit{ is a hard-core set} \}$

$H(n, k) =_{def} \{ S \subseteq P_{k+1}(n) : S \textit{ is a hard-core set} \}$

Obviously

$$| H(n, k) | = \sum_{r=1}^{\binom{n}{k+1}} | HC(n, k, r) | \qquad \bullet$$

For $| H(n, k) |$ it is easy to show that

$$2^{\binom{n}{k+1} - n + k} \leq | H(n, k) | \leq \binom{n}{k} 2^{\binom{n}{k+1} - n + k}$$

For the specific case $k = 1$ we obtain an asymptotically exact estimate.

Lemma 4.1: $\neg \exists\ S \subseteq P_2(n)$ *such that S covers exactly n-1 hard-core elements.*

Proof: Suppose, to the contrary, that S is such a set. Without loss of generality, let $\{ 1, 2, \ldots, n-1 \}$ be the hard-core elements covered by S. Then, from the definition of hard-core element, it follows that S must contain each of the sets $\{ i, n \}$ for all $1 \leq i \leq n-1$. But this means that S also covers the hard-core element n contradicting the assumption that S covered exactly $n-1$ hard-core elements. □

Lemma 4.2: *Let r, m, n*$\in$**N** *such that*

$$0 \leq m \leq n-2 \quad ; \quad mn - m(m+1)/2 \leq r < (m+1)n - (m+1)(m+2)/2.$$

For any $S \subseteq P_2(n)$ *such that* $|S| = r$*, S covers at most m distinct hard-core elements.*

Proof: (**Note:** The case $m = n-1$ gives $n(n-1)/2$ as the only permissible value of r. The only $S \subseteq P_2(n)$ containing this number of members is $P_2(n)$ itself, which clearly covers exactly n distinct hard-core elements.)

Let m, n, r be as in the Lemma statement and $S \subseteq P_2(n)$ with $|S| = r$. Suppose, that S covers t distinct hard-core elements: $\{y_1, y_2, \ldots, y_t\}$ say. Then for each $1 \le i \le t$ since S covers the hard-core element y_i it follows from the definition of hard-core that for all $x \in \{1, 2, \ldots, n\}$ with $y_i \neq x$ we have $\{x, y_i\} \in S$. To establish the lemma it suffices to show that $\forall\, t\ (0 \le t \le n-2)$

$$\left| \bigcup_{i=1}^{t} \bigcup_{\{x : 1 \le x \le n,\ x \neq y_i\}} \{\{x, y_i\}\} \right| = tn - t(t+1)/2 \tag{4.1}$$

We proceed by induction on t. The Inductive Base, t=0, is obvious since both the left and right-hand sides of inequality (4.1) are identically 0.

Inductively assume that the inequality in (4.1) holds for all values $< t \le n-2$ and prove that it holds for t. Let $S \subseteq P_2(n)$ be the smallest set that covers t hard-core elements, $\{y_1, \ldots, y_t\}$. Then S contains exactly the sets

$$\bigcup_{i=1}^{t} \bigcup_{j=i+1}^{n} \{\{y_i, y_j\}\}$$

The set $S - \bigcup_{i=t+1}^{n} \{\{y_t, y_i\}\}$ still covers the hard-core elements $\{y_1, \ldots, y_{t-1}\}$ and is a minimal such set. By the inductive hypothesis it contains exactly $(t-1)n - t(t-1)/2$ sets from $P_2(n)$. Hence S contains exactly $(t-1)n - t(t-1)/2 + (n-t) = tn - t(t+1)/2$ sets. This completes the Inductive Step.

The Lemma now follows easily since any S covering $m+1$ hard-core elements must contain at least $(m+1)n - (m+1)(m+2)/2$ sets, which exceeds the maximum number permitted by the range of r. □

Lemma 4.3:

$$|HC(n, 1, \binom{n}{2})| = \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} = 1$$

Proof: That $|HC(n, 1, \binom{n}{2})| = 1$ is immediate from the fact that there is only one subset of $P_2(n)$ containing $\binom{n}{2}$ sets, i.e. $P_2(n)$, and this covers n hard-core elements. The identity

$$\sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} = 1 \tag{4.2}$$

follows from the Binomial Theorem. □.

The identity described by (4.2) is used when estimating $|H(n, 1)|$ subsequently.

Lemma 4.4: $\forall\, r, m, n \in \mathbf{N}$ *such that*

$$0 \le m \le n-2 \quad ; \quad mn - m(m+1) \le r < (m+1)n - (m+1)(m+2)/2$$

it holds that

$$|HC(n, 1, r)| = \sum_{i=1}^{m} (-1)^{i+1} \binom{n}{i} \binom{\binom{n}{2} - ni + i(i+1)/2}{r - ni + i(i+1)/2}$$

Proof: Let r, m and n be as in the Lemma statement. For each i with $1 \le i \le m$ we define $P_r(i, n)$ to be the set of *ordered* sets of size r formed as follows:

> Each $T \in P_r(i, n)$ consists of two subsets: *Start*(T) *followed by Rest*(T). The ordering of sets within *Start*(T) is not significant; and the ordering of

sets within $Rest(T)$ is not significant. $Start(T)$ is formed by choosing i elements ($y_1, \ldots, y_i$, say) from $\{1, 2, \ldots, n\}$ and $Start(T)$ contains the $ni - i(i+1)/2$ sets from $P_2(n)$ needed to ensure that $Start(T)$ covers the i hard-core elements $y_1, \ldots, y_i$. $Rest(T)$ consists of a set of $r - ni + i(i+1)/2$ sets from $P_2(n)$ chosen from the $\left[\begin{array}{c} n \\ 2 \end{array} \right] - ni + i(i+1)/2$ sets that have not been used in $Start(T)$.

Notice that typically $T \in P_r(i, n)$ may cover more than the i hard-core elements included in $Start(T)$. In addition, when the ordering of $Start(T)$ followed by $Rest(T)$ is ignored, the set corresponding to T may be represented more than once in $P_r(i, n)$, e.g. suppose $i=1$ and $T = \langle c_1 ; c_2, c_3, \ldots, c_t, S \rangle$ where c_j denotes the sets from $P_2(n)$ needed to ensure that j is a hard-core element covered by T and S is a subset of $P_2(n)$ that does not create any new hard-core elements. Then T appears t times in $P_r(1, n)$, i.e. in each of the forms

$$\langle c_j ; c_1, \ldots, c_{j-1}, c_{j+1}, \ldots, c_t, S \rangle$$

Clearly, for all $1 \le i \le m$, we have

$$| P_r(i, n) | \;=\; \left[\begin{array}{c} n \\ i \end{array} \right] \left[\begin{array}{c} \left[\begin{array}{c} n \\ 2 \end{array} \right] - ni + i(i+1)/2 \\ r - ni + i(i+1)/2 \end{array} \right] \qquad (4.3)$$

We now define a partition of $P_r(i, n)$ into $m - i + 1$ sets E^i_j ($i \le j \le m$) by

$$E^i_j \;=\; \{\, T \in P_r(i, n) : T \textit{ covers exactly } j \textit{ hard-core elements} \,\}$$

Finally, for each k, with $1 \le k \le m$

$$S_k \;=\; \{\, S \subseteq P_2(n) : |S| = r \textit{ and } S \textit{ covers exactly } k \textit{ hard-core elements} \,\}$$

We have that:

$$| HC(n, 1, r) | \;=\; \sum_{i=1}^{m} | S_i | \quad ; \quad | P_r(i, n) | \;=\; \sum_{j=i}^{m} | E^i_j |$$

Now consider any set $Q \in S_t$. By the definition of $P_r(i, n)$ ordered sets containing exactly the same sets from $P_2(n)$ as are in Q occur in each $P_r(i, n)$ for $1 \le i \le t$. In particular there are *exactly* $\left[\begin{array}{c} t \\ i \end{array} \right]$ members of $P_r(i, n)$ containing the same sets from $P_2(n)$ as Q. To see this suppose $Q = \{ c_1, c_2, \ldots, c_t, S \}$ where c_j are the sets needed to ensure that Q covers a hard-core element y_j and S are the remaining sets in Q that do not contribute to the covering of one of the t hard-core elements. Then sets corresponding to Q are formed in $P_r(i, n)$ by choosing any subset of size i from the hard-core elements $\{ y_1, \ldots, y_t \}$; including the relevant c_j sets to form *Start*; and using the remaining $t - i$ core sets and the sets in S to form *Rest*.

It follows from the observations in the paragraph above that,

$$| E^i_t | \;=\; | S_t | \left[\begin{array}{c} t \\ i \end{array} \right] \qquad \forall\ 1 \le t \le m,\ 1 \le i \le t$$

Thus,

$$| P_r(i, n) | \;=\; \sum_{j=i}^{m} | E^i_j | \;=\; \sum_{j=i}^{m} | S_j | \left[\begin{array}{c} j \\ i \end{array} \right] \qquad (4.4)$$

So now consider the expression

$$\sum_{i=1}^{m} (-1)^{i+1} \mid P_r(i, n) \mid \tag{4.5}$$

From (4.4)

$$\begin{aligned}
\sum_{i=1}^{m} (-1)^{i+1} \mid P_r(i, n) \mid &= \sum_{i=1}^{m} (-1)^{i+1} \sum_{j=i}^{m} \mid S_j \mid \begin{bmatrix} j \\ i \end{bmatrix} &(4.6)\\
&= \sum_{j=1}^{m} \mid S_j \mid \Big(\sum_{i=1}^{j} (-1)^{i+1} \begin{bmatrix} j \\ i \end{bmatrix} \Big)
\end{aligned}$$

(This expression follows from the fact that in (4.6) the coefficient of $\mid S_j \mid$ is $\sum_{i=1}^{j} (-1)^{i+1} \begin{bmatrix} j \\ i \end{bmatrix}$.)

$$\begin{aligned}
&= \sum_{j=1}^{m} \mid S_j \mid \Big(- \sum_{i=0}^{j} (-1)^{i} \begin{bmatrix} j \\ i \end{bmatrix} + 1 \Big)\\
&= \sum_{j=1}^{m} \mid S_j \mid
\end{aligned}$$

(From the Binomial Theorem)

$$= \mid HC(n, 1, r) \mid$$

The Lemma now follows immediately from the size of $P_r(i, n)$ given by (4.3) and from the expansion of (4.5) above. □

Theorem 4.1: *Let* $\lambda(m) = mn - m(m+1)/2$ *where* $1 \le m \le n-1$. *Then*

$$\mid H(n, 1) \mid \;=\; \sum_{m=1}^{n-1} (-1)^{m+1} \begin{bmatrix} n \\ m \end{bmatrix} 2^{\begin{bmatrix} n \\ 2 \end{bmatrix} - \lambda(m)} \;+\; (-1)^{n+1}$$

Proof: We have that,

$$\mid H(n, 1) \mid \;=\; \sum_{r=n-1}^{\begin{bmatrix} n \\ 2 \end{bmatrix}} \mid HC(n, 1, r) \mid$$

From Lemma 4.3 and Lemma 4.4, this is equal to

$$\sum_{m=1}^{n-2} \sum_{r=\lambda(m)}^{\lambda(m+1)-1} \sum_{i=1}^{m} (-1)^{i+1} \begin{bmatrix} n \\ i \end{bmatrix} \begin{bmatrix} \begin{bmatrix} n \\ 2 \end{bmatrix} - \lambda(i) \\ r - \lambda(i) \end{bmatrix} + \sum_{i=1}^{n} (-1)^{i+1} \begin{bmatrix} n \\ i \end{bmatrix} \begin{bmatrix} \begin{bmatrix} n \\ 2 \end{bmatrix} - \lambda(i) \\ \begin{bmatrix} n \\ 2 \end{bmatrix} - \lambda(i) \end{bmatrix}$$

The term $(-1)^{i+1} \begin{bmatrix} n \\ i \end{bmatrix}$ occurs for each i $(1 \le i \le n)$. Furthermore, for $1 \le i \le n-2$ and each r such that $\lambda(i) \le r \le \begin{bmatrix} n \\ 2 \end{bmatrix}$, in the expression above there is exactly one occurrence of the summand

$$(-1)^{i+1} \begin{bmatrix} n \\ i \end{bmatrix} \begin{bmatrix} \begin{bmatrix} n \\ 2 \end{bmatrix} - \lambda(i) \\ r - \lambda(i) \end{bmatrix}$$

For $i = n-1$ and $i = n$ there are terms of the form

$$(-1)^n \binom{n}{n-1} \binom{\binom{n}{2} - \lambda(n-1)}{\binom{n}{2} - \lambda(n-1)} \quad ; \quad (-1)^{n+1} \binom{n}{n} \tag{4.7}$$

Collecting terms with a common coefficient of $(-1)^{i+1} \binom{n}{i}$ gives for each $1 \le i \le n-2$

$$\begin{aligned} \sum_{r=\lambda(i)}^{\binom{n}{2}} \binom{\binom{n}{2} - \lambda(i)}{r - \lambda(i)} &= \sum_{r=0}^{\binom{n}{2}-\lambda(i)} \binom{\binom{n}{2} - \lambda(i)}{r} \\ &= 2^{\binom{n}{2} - \lambda(i)} \end{aligned} \tag{4.8}$$

For the cases $i = n-1$ and $i = n$ the coefficients of $(-1)^{i+1} \binom{n}{i}$ are both identically 1. Hence

$$\begin{aligned} |\, H(n,\, 1,\, r)\, | &= \sum_{i=1}^{n} (-1)^{i+1} \binom{n}{i} (\, \textit{Coefficient from } (4.8)\,) \\ &= \sum_{i=1}^{n-1} (-1)^{i+1} \binom{n}{i} 2^{\binom{n}{2} - \lambda(i)} + (-1)^{n+1} \end{aligned}$$

as claimed. □

Corollary 4.1:

$$|\, H(n,\, 1)\, | \sim n\, 2^{\binom{n}{2} - n + 1}$$

Proof: In the expression for $|\, H(n, 1)\, |$, proved in Theorem 4.1, the dominant positive term occurs when $m = 1$ and the dominant negative term when $m = 2$. Hence

$$\begin{aligned} |\, H(n,\, 1)\, | &\sim n\, 2^{\binom{n}{2} - n + 1} - \binom{n}{2} 2^{\binom{n}{2} - 2n + 3} \\ &\sim n\, 2^{\binom{n}{2} - n + 1} \end{aligned}$$ □

Corollary 4.2: *The number of disjunctive hard-core functions in* $\tilde{Q}_{n,\,2}$ *is asymptotically equal to*

$$n\, 2^{\binom{n}{2} - n + 1}$$ □

5. The combinational complexity of functions built from variant sets

One of the important properties of Berkowitz' slice function transformation is the following fact: given the $n+1$ (T^n_k, T^n_{k+1})-variants of any $f \in B_n$, $\{ f_k \}$ say, (where $0 \le k \le n$, $T^n_0 \equiv 1$, $T^n_{n+1} \equiv 0$) the identity

$$f \equiv \bigvee_{k=0}^{n} (\, f_k \wedge (\, \neg\, T^n_{k+1}\,)\,)$$

holds. Thus, if f has superpolynomial combinational complexity, then we know that *some* slice function of f must have superpolynomial monotone complexity. In this section we show how given any $g \in M_n$ a sequence of n-argument monotone Boolean functions $< g_0, g_1, \ldots, g_r >$, which includes g, can be constructed with the properties that

P1. $g_0 \equiv 1$.

P2. $g_r \equiv 0$.

P3. $\forall\ 0 \le i < r,\ g_{i+1} < g_i$.

P4. $\forall\ 0 \le i < r,\ g_{i+1}$ is a ceiling of g_i.

P5. $\forall\ f \in B_n,\ f \equiv \bigvee_{k=0}^{r-1} (f \wedge g_k \vee g_{k+1}) \wedge (\neg g_{k+1})$.

Properties P4 and P5 establish that, provided the monotone complexity of each g_k is polynomial and r is bounded above by a polynomial in n, *if f* has superpolynomial combinational complexity then *at least one* (g_k, g_{k+1})-variant of f must have superpolynomial *monotone* complexity. In conjunction with Theorem 3.1, proving such a bound on the monotone complexity of the appropriate (g_k, g_{k+1})-variant of f would give the same size of bound on the combinational complexity of f.

Note: For the explicit construction developed below, it is possible that the condition 'r is bounded by a polynomial in n' is redundant, i.e. for all $g \in M_n$ this construction guarantees $r \le n^k$, for some k. At present, however, we have not been able to establish that such is the case.

Definition 5.1: The functionals $\Gamma : M_n \to M_n$ and $\Delta : M_n \to M_n$ are defined as

$$\Gamma(g)(\mathbf{X_n}) =_{def} \bigwedge_{q \in \mathbf{PC}(g)} \ \bigwedge_{x \in \mathbf{X_n} : x \nleq q} (x \vee q)$$

$$\Delta(g)(\mathbf{X_n}) =_{def} \bigvee_{p \in \mathbf{PI}(g)} \ \bigvee_{x \in \mathbf{X_n} : p \nleq x} (x \wedge p)$$

For $g \in M_n$, $\Gamma^k(g)$ and $\Delta^k(g)$ $(k \ge 0)$ denote the k-fold iterates of Γ and Δ respectively, i.e. $\Gamma^0(g) = \Delta^0(g) = g$ and $\Gamma^{k+1}(g) = \Gamma(\Gamma^k(g))$. We use the convention that $\Gamma(1) = 1$ and $\Delta(0) = 0$. Finally for $g \in M_n$,

$$\gamma(g) =_{def} \min \{\ k : \Gamma^k(g) \equiv 1\ \}$$

$$\delta(g) =_{def} \min \{\ k : \Delta^k(g) \equiv 0\ \} \qquad \bullet$$

With the convention that the empty disjunction is equivalent to 0 and the empty conjunction equal to 1 it is clear that $\gamma(g)$ (resp. $\delta(g)$) is well-defined for all $g \in M_n - \{0\}$ (resp. $g \in M_n - \{1\}$).

Lemma 5.1: $\forall\ g \in M_n\ Min(\Gamma(g)) \equiv Max(\Delta(g)) \equiv g$

Proof: Obvious. □

Definition 5.2: The functional $\psi : \mathbf{N} \times M_n \to M_n$ is defined by:

$$\psi(k, g) =_{def} \begin{cases} \Gamma^{\gamma(g)-k}(g) & \text{if} \quad 0 \le k \le \gamma(g) \\ \Delta^{k-\gamma(g)}(g) & \text{if} \quad \gamma(g) \le k \le \gamma(g) + \delta(g) \end{cases}$$

The functional $\Phi : \mathbf{N} \times M_n \times M_n \to M_n$ is given by

$$\Phi(k, g, f) =_{def} (f \wedge \psi(k, g)) \vee \psi(k+1, g) \qquad \bullet$$

Lemma 5.2:

$$\Phi(\gamma(g) - k, g, f) \equiv (f \wedge \Gamma^k(g)) \vee \Gamma^{k-1}(g) \qquad (1 \le k \le \gamma(g))$$

$$\Phi(\gamma(g)+k, g, f) \equiv (f \wedge \Delta^k(g)) \vee \Delta^{k+1}(g) \qquad (0 \le k < \delta(g))$$

Proof: Immediate from the definitions of Φ and ψ given above. □

Returning to our example function

$$f(x_1, x_2, x_3) = x_1 \vee x_2 \wedge x_3 \equiv (x_1 \vee x_2) \wedge (x_1 \vee x_3)$$

it is easily seen that, $\gamma(f)=2$, $\delta(f)=3$, and the sequence of 6 functions, $\psi(i, f)$ $(0 \le i \le 5)$ is

$$\psi(0, f) = 1 \quad ; \quad \psi(1, f) = x_1 \vee x_2 \vee x_{3\$} \quad ; \quad \psi(2, f) = f$$

$$\psi(3, f) = x_1 \wedge x_2 \vee x_1 \wedge x_3 \quad ; \quad \psi(4, f) = x_1 \wedge x_2 \wedge x_3 \quad ; \quad \psi(5, f) = 0$$

Theorem 5.1: $\forall\ g \in M_n$

P1. $\psi(0, g) \equiv 1.$

P2. $\psi(\gamma(g) + \delta(g), g) \equiv 0.$

P3. $\forall\ 0 \le k < \gamma(g)+\delta(g),\ \psi(k+1, g) < \psi(k, g).$

P4. $\forall\ 0 \le k < \gamma(g)+\delta(g),\ \psi(k+1, g)$ is a ceiling of $\psi(k, g)$.

P5 $\forall\ f \in B_n$

$$f \equiv \bigvee_{k=0}^{\gamma(g)+\delta(g)} \Phi(k, g, f) \wedge \overline{\psi(k+1, g)}$$

Proof:

P1: $\psi(0, g) = \Gamma^{\gamma(g)}(g) \equiv 1$ from the definition of $\gamma(g)$.

P2: $\psi(\gamma(g)+\delta(g), g) = \Delta^{\delta(g)}(g) \equiv 0$ from the definition of $\delta(g)$.

P3 and P4: First suppose that $0 \le k < \gamma(g)$ then

$$\psi(k+1, g) = \Gamma^{\gamma(g)-k-1}(g) = h$$

for some function, h say.

$$\psi(k, g) = \Gamma^{\gamma(g)-k}(g) = \Gamma(\Gamma^{\gamma(g)-k-1}(g)) = \Gamma(h)$$

From Lemma 5.1, we have that $Min(\Gamma(h)) = h$, hence h is a ceiling of $\Gamma(h)$. From Lemma 3.1 it follows that $h \le \Gamma(h)$. That the inequality is strict follows from the definition of Γ since $\Gamma(h)$ is the function such that every prime clause of h is a hard-core element covered by $\Gamma(h)$.

Now suppose that $\gamma(g) \le k < \gamma(g)+\delta(g)$. In this case

$$\psi(k, g) = \Delta^{k-\gamma(g)}(g) = h$$

for some function, h say.

$$\psi(k+1, g) = \Delta^{k+1-\gamma(g)}(g) = \Delta(\Delta^{k-\gamma(g)}(g)) = \Delta(h)$$

From Lemma 5.1, $Max(\Delta(h)) = h$ and hence $\Delta(h)$ is a ceiling of h. The result $\Delta(h) < h$ now follows from Lemma 3.1 and the construction of Δ.

P5: Let $f \in B_n$, and define $F \in B_n$ by

$$F \equiv \bigvee_{k=0}^{\gamma(g)+\delta(g)-1} \Phi(k, g, f) \wedge \overline{\psi(k+1, g)}$$

$$\equiv \bigvee_{k=0}^{\gamma(g)+\delta(g)-1} f \wedge \psi(k, g) \wedge \overline{\psi(k+1, g)}$$

First suppose that $F(\alpha)=1$ for some $\alpha \in \{0,1\}^n$. Then there must be some k with $0 \le k < \gamma(g)+\delta(g)$ for which

$$(f \wedge \psi(k, g) \wedge \overline{\psi(k+1, g)})(\alpha) = 1$$

and thus $f(\alpha)=1$ also.

On the other hand, suppose that $f(\alpha)=1$ for some $\alpha \in \{0,1\}^n$. To show that $F(\alpha)=1$ also it is sufficient to show that

$$\exists\, k \quad \psi(k, g)(\alpha)=1 \ ; \ \psi(k+1, g)(\alpha)=0. \tag{5.1}$$

From P1, P2, and P3 above we know that $\forall\ 0 < k < \gamma(g)+\delta(g)$

$$0 \equiv \psi(\gamma(g)+\delta(g), g) < \psi(k+1, g) < \psi(k, g) < \psi(0, g) \equiv 1. \tag{5.2}$$

Hence since $\psi(\gamma(g)+\delta(g), g)(\alpha) \neq \psi(0, g)(\alpha)$, it follows from (5.2) that (5.1) holds. This is enough to establish that $f(\alpha)=1 \Rightarrow F(\alpha)=1$ as required. □

6. Conclusions

In this paper we introduced the concept of a monotone Boolean function h being a ceiling of a monotone Boolean function g. It has been shown that if h is a ceiling of g then the (g,h)-variant of any monotone Boolean function f, i.e. the function $F=(f \wedge g) \vee h$ is such that the monotone and combinational complexities of F differ by at most an additive term of $n\,\mathbf{C^m}(h)$. We have described necessary and sufficient conditions for h to be a ceiling of g and characterised the cases for which the (g, h)-variant of f is not equivalent to h. In Section 4, some combinatorial estimates of the number of pairs (g, h) within a specific class, having the property that the (g,h)-variant is not identical to h, were obtained. The exact counting argument given here does not extend to larger classes of monotone Boolean function and it remains an open combinatorial problem to prove exact asymptotic estimates for these cases. Finally, in Section 5, we have shown how any $f \in B_n$ may be efficiently constructed given a sequence of appropriate (g, h)-variants of f. In consequence it can be seen that the slice function transformation of [Berkowitz 1983], represents a special case of the more general translation classes given by ceilings and variants as described above.

7. References

[Alon and Boppana 1986] Alon, N., Boppana, R.: "The monotone circuit complexity of Boolean functions"; Combinatorica, 7, (1986), 1-22

[Andreev 1985] Andreev, A.E.: "A method of proving lower bounds on the complexity of monotone Boolean functions"; Doklady Akademii-Nauk SSSR, 282, (1985), 1033-1037

[Berkowitz 1983] Berkowitz, S.: "On some relationships between monotone and non-monotone circuit complexity"; Technical Report, Univ. of Toronto, (1982)

[Dunne 1984] Dunne, P.E.: "Techniques for the analysis of monotone Boolean networks"; Ph.D Dissertation, Univ. of Warwick, (1984)

[Dunne 1986] Dunne, P.E.: "The complexity of central slice functions"; Theoretical Computer Science, 44, (1986), 247-257

[Dunne 1988] Dunne, P.E.: "The complexity of Boolean networks"; Academic Press, (1988)

[Dunne 1989] Dunne, P.E.: "On monotone simulations of non-monotone networks"; Theoretical Computer Science, 66, (1989), 15-25

[Dunne 1992] Dunne, P.E.: "Relationships between monotone and non-monotone network complexity"; In: Paterson, M.S. (Ed) Boolean Function Complexity, (London Math. Soc. Lecture Note Series, 169) Cambridge University Press, (1992), 1-24

[Fischer and Pippenger 1979] Fischer, M.J., Pippenger, N.: "Relations among complexity measures"; Jnl. of the A.C.M., 26, (1979), 361-381

[Razborov 1985] Razborov, A.A.: "A lower bound on the monotone complexity of the logical permanent"; Matem. Zametki, 37, (1985), 887-901

[Schnorr 1976] Schnorr, C.P.: "The network complexity and Turing machine complexity of finite functions"; Acta Inf., 7, (1976), 95-107

[Shannon 1949] Shannon, C.E.: "The synthesis of two-terminal switching circuits", Bell Syst. Tech. Jnl., 28, (1949), 59-98

[Valiant 1986] Valiant, L.G.: "Negation is powerless for Boolean slice functions"; S.I.A.M. Jnl. on Computing, 15, (1986), 531-535