A NOTE ON THE RELATIONS BETWEEN CRITICAL AND SENSITIVE COMPLEXITY

Ingo Wegener*, Fachbereich Informatik, LS II, Universität Dortmund
4600 Dortmund, Fed. Rep. of Germany

and

Laszlo Zádori, Dept. of Mathematics, Statistics and Computer Science,
University of Illinois at Chicago, Chicago, Ill. 60680, USA

Abstract

Sensitive and critical complexity of a Boolean function f are measures for the PRAM and WRAM complexity of f. It is proved that the maximum and even the minimum sensitive complexity of f can be much larger than its critical complexity. This result answers the last open questions on the relations between these and other combinatorical complexity measures.

----

* Supported in part by DFG grants We 1066/1-2 and Me 872/1-1

1. INTRODUCTION

Cook, Dwork and Reischuk [2] defined the critical complexity c(f) of a Boolean function f. For $a \in \{0,1\}^n$, c(f,a) is the number of neighbors b of a (Hamming distance between a and b is 1) such that $f(b) \neq f(a)$. Then c(f) is the maximum of all c(f,a) where $a \in \{0,1\}^n$.

Vishkin and Wigderson [6] defined the sensitive complexity s(f,a) of a Boolean function f at input a. Let d(f,a) be the dimension of the largest subcube of $\{0,1\}^n$ such that the subcube contains a and f is constant on this subcube. Then s(f,a) := n-d(f,a). The following definition is equivalent (see Wegener [9]). If f(a) = 1, s(f,a) is the length of the shortest prime implicant covering a. If f(a) = 0, one has to consider prime clauses. $s_{max}(f)$ is the maximum of all s(f,a), and $s_{min}(f)$ is the minimum of all s(f,a).

These complexity measures are motivated by the lower bound techniques for PRAMs and WRAMS due to Cook, Dwork and Reischuk [2] and Vishkin and Wigderson [6]. These measures have been used also for other lower bound techniques for parallel computers. Properties of these complexity measures have been studied e.g. by Schürfeld and Wegener [3], Simon [4], Turàn [5] and Wegener [8]. Relations between these and other complexity measures have been investigated by Bublitz, Schürfeld, Voigt and Wegener [1]. We mention the known relations between c, s_{max} and s_{min}.

- $c(f) \le s_{max}(f)$ and $s_{min}(f) \le s_{max}(f)$ for all Boolean functions $f \in B_n$.
- $c(f) = s_{max}(f)$ for all monotone functions $f \in M_n$.
- $s_{min}(f) \le c(f)$ for all monotone or symmetric functions $f \in M_n \cup S_n$.

We also mention the largest known differences between these complexity measures.

- $s_{min}(f_n) = 1$ but $c(f_n) = s_{max}(f_n) = n$ for $f_n(x) = x_1 \wedge \dots \wedge x_n$.
- $s_{min}(f_n) / c(f_n) = 5/3$ for some explicitly defined functions f_n on n = 6k variables.
- $s_{max}(f_n) / c(f_n) = (n-1) / (\lfloor n/2 \rfloor + 2) \to 2$ as $n \to \infty$ for some explicitly defined symmetric functions $f_n \in S_n$.

The last two results are unsatisfying, since the best known upper bound on $s_{min}(f_n)/c(f_n)$ or $s_{max}(f_n)/c(f_n)$ is of size n/log n. If $s_{max}(f)$ and c(f) were of the same size, we could replace c(f) by $s_{max}(f)$ and could work with the more natural measure of sensitive complexity, which is defined by the length of prime implicants and prime clauses.

In the following we prove that $s_{min}(f_n)/c(f_n)$ and $s_{max}(f_n)/c(f_n)$ may be rather large, namely of size $n^{0.29}$. In section 2 we prove some results on the critical and sensitive complexity of compound Boolean functions. In section 3 we apply these results and construct explicitly Boolean functions whose sensitive complexity is much larger than their critical complexity.

2. THE CRITICAL AND SENSITIVE COMPLEXITY OF COMPOUND FUNCTIONS

We introduce the notion of compound Boolean functions. Let $f \in B_k$ and $g \in B_m$. Then fg is the (compound) function on km variables, grouped into k disjoint blocks $x^1,\dots,x^k$ where $x^i = (x^i_1,\dots,x^i_m)$ and
$fg(x) := f(g(x^1),\dots,g(x^k))$.

<u>Lemma 1:</u> $c(fg) \le c(f)c(g)$.

<u>Proof:</u> Let a be some input for fg. $b_i := g(a^i)$ and $b = (b_1,\dots,b_k)$. There are at most c(f) neighbors b' of b where $f(b') \ne f(b)$. Let $b^{(j)}$ be the neighbor of b which differs from b at position j. If $f(b^{(j)}) = f(b)$, a change of one bit in a^j cannot cause a change of the output. Therefore, only changes in one of the at most c(f) blocks a^j where $f(b^{(j)}) \ne f(b)$ are interesting. There are at most c(g) neighbors $\tilde{a}^j$ of a^j such that $g(\tilde{a}^j) \ne g(a^j)$. Hence, fg is at most c(f)c(g)-critical.

□

<u>Lemma 2:</u> $s_{min}(fg) \ge s_{min}(f)\, s_{min}(g)$.

<u>Proof:</u> We use the following interpretation of $s_{min}(h)$. $s_{min}(h)$ is the minimal number of variables such that replacing the variables by appropriate constants yields a constant subfunction of h. If some block contains less than $s_{min}(g)$ constant inputs, $g(x^i)$ is not replaced by a constant. If less than $s_{min}(f)$

block outputs $g(x^i)$ are replaced by constants, fg is not replaced by a constant. Hence, at least $s_{min}(f)\ s_{min}(g)$ variables have to be replaced by constants in order to obtain a constant subfunction.

□

The estimation of $s_{max}(fg)$ is a little bit more difficult. Let $s_{max}^{PI}(h)$ and $s_{max}^{PC}(h)$ be the maximum of all s(h,a) where h(a) = 1 and h(a) = 0 resp. We use the following interpretation of s(h,a). s(h,a) is the minimal number of variables x_i which have to be replaced by the corresponding a_i-value such that we obtain a constant subfunction of h.

We like to estimate for some $b = (b_1,\dots,b_k)$ the maximum of all s(fg,a) where $b_i = g(a^i)$. W.l.o.g. f(b) = 1. Let $t_1,\dots,t_r$ be the prime implicants of f covering b and let i_ℓ and j_ℓ be the number of positive and negative resp. literals in t_ℓ. In order to obtain a constant subfunction of fg by replacing variables in such a way by constants that the i-th g-function can compute b_i, it is necessary to replace, for some ℓ, the appropriate i_ℓ g-functions by 1 and the appropriate j_ℓ g-functions by 0. For this purpose we can choose inputs a^i such that we have to replace $i_\ell s_{max}^{PI}(g) + j_\ell s_{max}^{PC}(g)$ variables x_t^s by a_t^s. Similar considerations hold for vectors b where f(b) = 0. Hence, we have proved the following lemma.

<u>Lemma 3:</u> For $b \in f^{-1}(1)$ $[b \in f^{-1}(0)]$ let r(b) be the number of prime implicants [prime clauses] covering b and let $i_\ell(b)$ and $j_\ell(b)$ be the number of positive and negative resp. literals in the ℓ-th prime implicant [prime clause] covering b. Then

$$s_{max}^{PI}(fg) \geq \max_{b \in f^{-1}(1)} \min_{1 \leq \ell \leq r(b)} \{i_\ell(b)\, s_{max}^{PI}(g) + j_\ell(b)\, s_{max}^{PC}(g)\}$$

and

$$s_{max}^{PC}(fg) \geq \max_{b \in f^{-1}(0)} \min_{1 \leq \ell \leq r(b)} \{i_\ell(b)\, s_{max}^{PC}(g) + j_\ell(b)\, s_{max}^{PI}(g)\}.$$

We finish the section with a remark. If f is the logical-and of two inputs, Voigt and Wegener [7] have shown that one does in general not obtain a minimal polynomial for fg by applying the law of distributivity to the conjunction of minimal polynomials for $g(x^1)$ and $g(x^2)$, Nevertheless, by Lemma 3, $s_{max}^{PI}(fg) = 2 \cdot s_{max}^{PI}(g)$.

3. FUNCTIONS WHOSE SENSITIVE COMPLEXITY IS MUCH LARGER THAN THEIR CRITICAL COMPLEXITY

We now apply the results of the last section in order to obtain functions whose sensitive complexity is much larger than their critical complexity. We have to start with a function $f \in B_k$ whose sensitive complexity is larger than its critical complexity. Let $f^1 := f$ and $f^\ell := f f^{\ell-1}$ be the compound function on k^ℓ variables. The largest known value for $s_{max}(f)/c(f)$ has been obtained for the symmetric function $f \in S_k$ computing 1 iff the number of ones in the input is $\lfloor k/2 \rfloor$ or $\lfloor k/2 \rfloor + 1$.

Obviously, $c(f^1) = \lfloor k/2 \rfloor + 2$ and, by Lemma 1, $c(f^\ell) \le (\lfloor k/2 \rfloor + 2)^\ell$. Moreover, $s^{PI}_{max}(f^1) = k-1$ and all prime implicants consist of $\lfloor k/2 \rfloor$ positive literals and $\lceil k/2 \rceil - 1$ negative literals. Finally, $s^{PC}_{max}(f^1) = \lfloor k/2 \rfloor + 2$ and there are prime clauses consisting of $\lfloor k/2 \rfloor + 2$ negative literals and prime clauses consisting of $\lceil k/2 \rceil + 1$ positive literals. No $a \in f^{-1}(0)$ is covered by both types of prime clauses. Hence, by Lemma 3,

$$s^{PI}_{max}(f^\ell) \ge \lfloor k/2 \rfloor \, s^{PI}_{max}(f^{\ell-1}) + (\lceil k/2 \rceil - 1)\, s^{PC}_{max}(f^{\ell-1}) \text{ and}$$

$$s^{PC}_{max}(f^\ell) \ge \max\{(\lfloor k/2 \rfloor + 2)\, s^{PI}_{max}(f^{\ell-1}), (\lceil k/2 \rceil + 1)\, s^{PC}_{max}(f^{\ell-1})\}$$

$$= (\lfloor k/2 \rfloor + 2)\, s^{PI}_{max}(f^{\ell-1}).$$

We apply the second inequality to $s^{PC}_{max}(f^{\ell-1})$ and insert the result into the first inequality. Then, for even k, $s^{PI}_{max}(f^\ell) \ge \frac{k}{2} s^{PI}_{max}(f^{\ell-1}) + (\frac{k}{2})^2 s^{PI}_{max}(f^{\ell-2})$.

Let $S_\ell := s^{PI}_{max}(f^\ell)\,(\frac{2}{k})^\ell$. Then

$S_1 \ge 2 - \frac{2}{k}$, $S_2 \ge 3 + \frac{2}{k} - \frac{12}{k^2}$ and $S_\ell \ge S_{\ell-1} + S_{\ell-2}$, which is the Fibonacci recursion. Hence, it is easy to prove that $S_\ell \ge \mathrm{Fib}(\ell+2)$ for $\ell \ge 4$ and that for all ℓ and $c := \frac{1}{2}(1 + \sqrt{5})$

$$s_{max}(f^\ell) = s^{PI}_{max}(f^\ell) \ge (\tfrac{ck}{2})^\ell.$$

Altogether,
$s_{max}(f^\ell)/c(f^\ell) \geq (c(1 - \frac{4}{k+4}))^\ell$.

f^ℓ is defined on $n = k^\ell$ variables, hence $\ell = \frac{\log n}{\log k}$. In order to maximize the quotient of maximal sensitive and critical complexity, we have to maximize $(c(1 - \frac{4}{k+4}))^{1/\log k}$. The optimal value for k is 34. Then the quotient of maximal sensitive and critical complexity grows as fast as

$$(\frac{1}{2}(1 + \sqrt{5})\frac{34}{38})^{(\log n)/\log 34} = n^{[\log(\frac{17}{38}(1 + \sqrt{5}))]/\log 34} \approx n^{0.105}.$$

The starting point of this consideration is not as good as it seemed at the first glance. For 34 variables $s_{max}(f)/c(f) = \frac{33}{19} \approx 1.737$. We can prove better results. We know (see [1]) a function g on 6 variables such that $s_{min}(g) = 5$ and $c(g) = 3$. By Lemma 1 and Lemma 2, $s_{min}(g^\ell)/c(g^\ell) \geq (\frac{5}{3})^\ell$.

The number of variables of g^ℓ is $n = 6^\ell$. Hence,

$$s_{min}(g^\ell)/c(g^\ell) \geq n^{(\log 5 - \log 3)/\log 6} \approx n^{0.285}.$$

We obtain the best result by starting with the smallest example. Let $h \in B_4$ be defined by the following Karnaugh diagram.

h	00	01	11	10
00	0	1	1	1
01	0	0	0	1
11	1	1	0	1
10	0	1	0	0

Then $s_{min}(h) = 3$ and $c(h) = 2$. By Lemma 1 and Lemma 2,

$$s_{min}(h^\ell)/c(h^\ell) \geq (\frac{3}{2})^\ell.$$

The number of variables of h^ℓ is $n = 4^\ell$. Hence, we have shown our main result.

<u>Theorem:</u> There are explicitly defined Boolean functions h^ℓ on $n = 4^\ell$ variables such that

$$s_{max}(h^\ell)/c(h^\ell) \geq s_{min}(h^\ell)/c(h^\ell) \geq n^{(\log 3-1)/2} \geq n^{0.29248}.$$

It has been proved by Simon [4] that $c(f) = \Omega(\log n)$ for all nondegenerate functions f. Hence, $s_{max}(f)/c(f)$ cannot grow faster than n/log n and our result is not very far from optimal.

There is still an open problem. Are there Boolean functions f such that $s_{max}(f)$ grows faster than any polynomial in c(f)? If this were not the case, the lower bound of Cook, Dwork and Reischuk [2] could be expressed as $\Omega(\log s_{max}(f))$ and all the mentioned lower bounds for parallel computers would depend only on the sensitive complexity of Boolean functions. The only functions, for which it is known that the critical complexity is very small, are types of addressing functions. For these functions also the sensitive complexity is small.

Acknowledgment

We like to thank György Turán who established the contact between us.

References

[1] S.Bublitz/U.Schürfeld/B.Voigt/I.Wegener:
Properties of complexity measures for PRAMs and WRAMs, Theoretical Computer Science 48, 53-73, 1986.

[2] S.Cook/C.Dwork/R.Reischuk:
Upper and lower time bounds for parallel random access machines without simultaneous writes, SIAM J. on Computing 15, 87-97, 1986.

[3] U.Schürfeld/I.Wegener:
On the CREW PRAM complexity of Boolean functions, Parallel Computing 85, Eds. Feilmeier, Joubert,Schendel,Elsevier Publ., 247-252, 1986.

[4] H.U.Simon:
A tight $\Omega(\log \log n)$ bound on the time for parallel RAMs to compute nondegenerate Boolean functions, FCT '83, Lecture Notes in Computer Science 158,439-444, 1983.

[5] G.Turàn:
The critical complexity of graph properties, Information Processing Letters 18, 151-153, 1984.

[6] U.Vishkin/A.Wigderson:
Trade-offs between depth and width in parallel computation, SIAM J. on Computing 14, 303-314, 1985.

[7] B.Voigt/I.Wegener:
Minimal polynomials for the conjunction of functions on disjoint variables can be very simple, Techn. Rep., Univ. Dortmund, 1988

[8] I.Wegener:
The critical complexity of all (monotone) Boolean functions and monotone graph properties, Information and Control 67, 212-222, 1985.

[9] I.Wegener:
The complexity of Boolean functions, Wiley-Teubner, 1987.