On the Readability of Monotone Boolean Formulae

Khaled Elbassioni[1], Kazuhisa Makino[2], Imran Rauf[1]

[1] Max-Planck-Institut für Informatik, Saarbrücken, Germany
{elbassio,irauf}@mpi-inf.mpg.de
[2] Department of Mathematical Informatics, University of Tokyo, Tokyo, Japan
makino@mist.i.u-tokyo.ac.jp

Abstract. Golumbic et al. [Discrete Applied Mathematics 154(2006) 1465-1477] defined the *readability* of a monotone Boolean function f to be the minimum integer k such that there exists an $\wedge - \vee$-formula equivalent to f in which each variable appears at most k times. They asked whether there exists a polynomial-time algorithm, which given a monotone Boolean function f, in CNF or DNF form, checks whether f is a read-k function, for a fixed k. In this paper, we partially answer this question already for $k = 2$ by showing that it is NP-hard to decide if a given monotone formula represents a read-twice function. It follows also from our reduction that it is NP-hard to approximate the readability of a given monotone Boolean function $f : \{0,1\}^n \to \{0,1\}$ within a factor of $\mathcal{O}(n)$. We also give tight sublinear upper bounds on the readability of a monotone Boolean function given in CNF (or DNF) form, parameterized by the number of terms in the CNF and the maximum size in each term, or more generally the maximum number of variables in the intersection of any constant number of terms. When the variables of the DNF can be ordered so that each term consists of a set of consecutive variables, we give much tighter polylogarithmic bounds on the readability.

1 Introduction

Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone Boolean function, i. e., for any $x, x' \in \{0,1\}^n$, $x' \geq x$ implies $f(x') \geq f(x)$. One property of such functions is that they can be represented by negation-free Boolean formulae. A *minterm* (*maxterm*) of monotone Boolean function $f(x_1, \ldots, x_n)$ is a minimal set of variables which, if assigned the value 1 (resp., value 0), forces the function to take the value 1 (resp., value 0) regardless of the values assigned to the remaining variables. It is well-known that the irredundant (i. e., no term contains another) *disjunctive normal form* (DNF) and *conjunctive normal form* (CNF) of monotone Boolean function f consist respectively of all of its minterms and maxterms (cf. [Weg87]).

A monotone read-k formula is a Boolean formula over the operators $\{\vee, \wedge\}$ in which each variable occurs at most k times. The readability of f is the minimum k such that f can be represented by a monotone read-k formula. We also call f a read-k function when it has readability k. Finding the readability of an arbitrary

Boolean function and computing a formula which achieves this readability has applications in circuit design among others and therefore is one of the earliest problems considered in Computer Science [GMR06].

Given a monotone Boolean function in one of the normal forms (CNF/DNF), a complete combinatorial characterization for it to be read-once was given by Gurvich [Gur77]. A polynomial-time algorithm based on this criterion is given by Golumbic et al. [GMR06] to decide whether a given CNF or DNF is read-once. The algorithm also computes the unique read-once representation when a read-once function is given as input. For $k \geq 2$, no characterization is known for a given monotone Boolean CNF or DNF to be read-k, and in fact, Golumbic et al. asked in [GMR06] whether there exists a polynomial-time algorithm, which given a (normal) monotone Boolean function f in CNF or DNF form, checks whether f is a a read-k function, for a fixed k.

The case when the function is given by an oracle has also been considered in the machine learning community. It is shown in [AHK93] that given a read-once function by a membership oracle, we can compute its read-once representation in polynomial time. However, the correctness of the algorithm is based on the assumption that the function provided as an oracle is read-once. If its not read-once then the algorithm terminates with incorrect output.

In this paper, we show that, given an $\wedge - \vee$-formula, it is NP-hard to check if it represents a read-twice function f. This partially answers the question of Golumbic et al. [GMR06], but leaves open the case when f is given by the CNF or DNF normal form. It follows also from our reduction that it is NP-hard to approximate the readability of a given monotone Boolean function $f : \{0,1\}^n \rightarrow \{0,1\}$ within a factor of $\mathcal{O}(n)$.

It follows from a result in [Weg87] that almost all monotone Boolean functions on n variables, in which each minterm has size exactly k, have readability $\Omega(n^{k-1} \log^{-1} n)$. Assuming that the function is given by its irredundant DNF (or CNF) of m minterms, this implies a lower bound of $\tilde{\Omega}(m^{1-\frac{1}{k}})$ on the readability. This naturally raises the question whether this bound is tight, i.e for any monotone CNF formula of m terms, there exists an equivalent read-$O(m^{1-\frac{1}{k}})$ representation. In this paper, we show that this indeed the case, and moreover that such a representation can be found in polynomial time. In fact, we prove a more general result. For integers $p, q > 0$, let us say a monotone CNF f has (p, q)-bounded intersection [KBEG07] if every p terms intersect in at most q variables. We show that any such CNF has read-$O((p + q - 1)m^{1-\frac{1}{q+1}})$ representation which can be found in polynomial-time. Confronted with this almost tight sublinear bound on readability, an interesting question is whether it can be improved for interesting special cases. For the class of interval DNF's, i.e. those for which there is an ordering on the variables such that each term contains only consecutive variables in that ordering, we show that readability is at most $O(\log^2 m)$.

The paper is organized as follows. In the next section, we point out that the characterization of [Gur77] for read-once functions does not carry over to read-k functions already for $k = 2$. In Section 3, we present upper bounds on the readability of some classes monotone Boolean DNF (resp. CNF) that depends only on the number of terms in the normal form. In Section 4 we show that finding the readability in general is hard when the input formula is not a DNF or CNF. We also give an $\mathcal{O}(n)$ inapproximability result in this case.

2 On Generalization of Read-once Functions

An elegant characterization of read-once functions is provided by the following theorem of Gurvich.

Theorem 1 ([Gur77]). *For any monotone Boolean function f the following two statements are equivalent: (i) f is read-once. (ii) Every minterm and maxterm of f intersect in exactly $c = 1$ variable.*

However, this result does not generalizes to read-twice functions as the following example shows. Consider the read-twice formula

$$g(x_1, \ldots, x_n, y_1, \ldots, y_n) = \bigwedge_{1 \le i \le n} (x_i \vee y_i) \bigwedge (x_1 \vee \ldots \vee x_n).$$

It is easy to see that the g has a minterm $x_1 \ldots x_n$ which intersects with the maxterm $(x_1 \vee \ldots \vee x_n)$ in n variables. Hence hypergraphs corresponding to read-twice functions do not necessarily satisfy the generalization of Condition (ii) of Theorem 1 for any constant $c > 1$. Conversely, any such generalization is also not sufficient for a function to be read-c, as implied by the following result on the shortest possible size of k-homogeneous DNF where the size of each term is exactly k (and hence each minterm and maxterm intersect in at most k).

Theorem 2 (cf. [Weg87]). *For an integer k, let $\mathscr{H}_k^n$ be the class of monotone Boolean functions on n variables such that size of every minterm is exactly k. The monotone formula size of almost all $h \in \mathscr{H}_k$ is $\Omega(n^k \log^{-1} n)$.*

Theorem 2 implies that the readability of almost all $h \in \mathscr{H}_k$ is $\Omega(n^{k-1} \log^{-1} n)$, since otherwise the formula achieving a smaller readability has smaller then shortest possible size.

3 Upper Bounds

In this section, we consider various classes of monotone Boolean DNF's and give upper bounds on their readability. First we consider Interval DNF's whose terms correspond to consecutive variables, given some ordering on variables. Next, we consider (p, q)-intersecting DNF where every p of its terms intersect in at most q variables and give an almost tight upper bound on their readability. Finally, we consider a special case of the latter class, namely k-DNF, where the size of each term is bounded by k and again give a tight upper bound on their readability. Even though we get the same upper bound implied by the more general case, the formula computed by our algorithm has only depth 3 in this case.

In our description of the algorithms, we use set-theoretic notations to describe various operations on the structure of DNF's. In this sense, we treat the DNF $\phi = \bigvee t_i$ as its corresponding hypergraph $\{t_i \mid t_i \text{ is a term in } \phi\}$. For example, we write $t \in \phi$ when t is term of ϕ and similarly by $x \in t$ we mean that the term t contains variable x. Let us denote the *degree* of a variable in ϕ by $\deg_\phi(x)$, which is the number of terms in ϕ containing $x \in V$. For a Boolean formula f and a literal x (resp. set of literals S) in f, we denote by $f \mid_{x=1}$ (resp. $f \mid_{S=1}$) the resulting f after replacing every occurrence of x (resp. $x \in S$) in f with 1.

3.1 Interval DNF

A monotone Boolean DNF $\mathcal{I} = \bigvee_{I \in \mathcal{I}} \bigwedge_{x \in I} x$ is called interval DNF if there is an ordering of variables $V = \{x_1, x_2, \ldots, x_n\}$ such that each $I \in \mathcal{I}$ contains only consecutive elements from the ordering. We show that an interval DNF containing m terms is $\mathcal{O}(\log^2 m)$-readable. For a variable $x_j \in V$, let $\mathcal{I}_{<x_j} = \{I \in \mathcal{I} : I \subseteq \{x_1, \ldots, x_{j-1}\}\}$, $\mathcal{I}_{>x_j} = \{I \in \mathcal{I} : I \subseteq \{x_{j+1}, \ldots, x_n\}\}$ and $\mathcal{I}_{\ni x_j} = \{I \in \mathcal{I} : x_j \in I\}$. For a term $I = x_i x_{i+1} \ldots x_j$ in interval DNF $\mathcal{I}$, we call x_i and x_j its left and right end-points, and denote them with $\mathrm{L}(I)$ and $\mathrm{R}(I)$ respectively. We also denote the first (resp. last) term in the ordering of terms of $\mathcal{I}$ with respect to their left end point as $\mathrm{first}(\mathcal{I})$ and $\mathrm{last}(\mathcal{I})$ respectively.

The algorithm is given in Figure 1. It proceeds by choosing a variable x_j such that at most half of the intervals are completely on the left ($\mathcal{I}_{<x_j}$) and half on the right ($\mathcal{I}_{>x_j}$). The formulae for $\mathcal{I}_{<x_j}$ and $\mathcal{I}_{>x_j}$ are computed recursively and the remaining terms in $\mathcal{I}_{\ni x_j}$ are divided into two halves ($\mathcal{I}_1$ and $\mathcal{I}_2$) by considering them in order with respect to their left end-point. The algorithm then factors out common variables from $\mathcal{I}_1$ and $\mathcal{I}_2$ and computes their equivalent formulae recursively.

Theorem 3. *Let $\mathcal{I}$ be an irredundant interval DNF containing m terms. Then $\mathcal{I}$ is $\mathcal{O}(\log^2 m)$-readable.*

Proof. We show that the procedure REDUCE1($\mathcal{I}$) returns a formula with $\mathcal{O}(\log^2 m)$-readability given an interval DNF. Let $r_1(m)$ and $r_2(m)$ be the readability of the formulae generated by the procedures REDUCE1($\mathcal{I}$) and REDUCE2($\mathcal{I}$), respectively, when given an interval DNF $\mathcal{I}$ containing m terms as input. Let x_j be the variable chosen in Step 2 of the algorithm. Since the subproblems $\mathcal{I}_{<x_j}$ and $\mathcal{I}_{>x_j}$ are disjoint and have size at most $m/2$, the recurrence for readability computed by REDUCE1($\mathcal{I}$) is $r_1(m) \leq r_1(m/2) + r_2(m)$.

Similarly, given an intersecting interval DNF $\mathcal{I}$ the procedure REDUCE2($\mathcal{I}$) divides the problem into subproblems $\mathcal{I}_1$ and $\mathcal{I}_2$ respectively. Note that the subproblems in the recursive call i.e. $(\mathcal{I}_1 \setminus \{\mathrm{first}(\mathcal{I}_1), \mathrm{last}(\mathcal{I}_1\}) \mid_{\phi_1=1}$ and $(\mathcal{I}_2 \setminus \{\mathrm{first}(\mathcal{I}_2), \mathrm{last}(\mathcal{I}_2)\}) \mid_{\phi_2=1}$ are again intersecting since $\mathcal{I}_1$ and $\mathcal{I}_2$ are irredundant. For calculating the readability of the formula computed by REDUCE2($\mathcal{I}$), consider the case when a variable x_i occurs in both subproblems. We show that if x_i does not occur in ϕ_1 (resp. ϕ_2) then it is necessarily the case that it appears in ϕ_2 (resp. ϕ_1) and thus occurring only once in at least one of the

Procedure REDUCE1($\mathcal{I}$)**:**

Input: A monotone Boolean interval DNF $\mathcal{I} = \bigvee_{j=1}^{m} I_j$ on variables ordering $x_1, \ldots, x_n$

Output: A $\mathcal{O}(\log^2 m)$ readable formula ψ equivalent to $\mathcal{I}$

1. **if** $|\mathcal{I}| \leq 2$ **then return** the read-once formula representing $\mathcal{I}$
2. Let $x_j \in V$ such that $|\mathcal{I}_{<x_j}| \leq \frac{m}{2}$ and $|\mathcal{I}_{>x_j}| \leq \frac{m}{2}$
3. **return** REDUCE1($\mathcal{I}_{<x_j}$) $\bigvee$ REDUCE2($\mathcal{I}_{\ni x_j}$) $\bigvee$ REDUCE1($\mathcal{I}_{>x_j}$)

Procedure REDUCE2($\mathcal{I}$)**:**

Input: A monotone Boolean interval DNF $\mathcal{I}$ s.t. $\bigcap_{j=1}^{m} I_j \neq \emptyset$

Output: A $\mathcal{O}(\log m)$ readable formula ψ equivalent to $\mathcal{I}$

1. **if** $|\mathcal{I}| \leq 2$ **then return** the read-once formula representing $\mathcal{I}$
2. Consider terms in $\mathcal{I}$ in order of their left end points, let $\mathcal{I}_1$ (resp. $\mathcal{I}_2$) be first half (resp. remaining half) elements of $\mathcal{I}$.
3. Let ϕ_1 (resp. ϕ_2) be maximum set of variables that occur in every term of $\mathcal{I}_1$ (resp. $\mathcal{I}_2$)
4. $t_1 := \text{first}(\mathcal{I}_1),\ t_2 := \text{last}(\mathcal{I}_1),\ t_3 := \text{first}(\mathcal{I}_2),\ t_4 := \text{last}(\mathcal{I}_2)$
5. $\psi_1 = \text{REDUCE2}((\mathcal{I}_1 \setminus \{t_1, t_2\}) \mid_{\phi_1=1}), \quad \psi_2 = \text{REDUCE2}((\mathcal{I}_2 \setminus \{t_3, t_4\}) \mid_{\phi_2=1})$
6. **return** $(\phi_1 \wedge (\psi_1 \vee t_1 \mid_{\phi_1=1} \vee t_2 \mid_{\phi_1=1})) \bigvee (\phi_2 \wedge (\psi_2 \vee t_3 \mid_{\phi_2=1} \vee t_4 \mid_{\phi_2=1}))$

Fig. 1. An algorithm to find an $\mathcal{O}(\log^2 m)$-readable formula for interval DNF consisting of m intersecting terms

subproblems. Note that since $\mathcal{I}$ is irredundant, the set ϕ_2 forms the interval $[\text{L}(\text{last}(\mathcal{I}_2)), \text{R}(\text{first}(\mathcal{I}_2))]$. Also observe that since x_i occurs in both subproblems and not in ϕ_1, it must lie in the interval $[\text{R}(\text{first}(\mathcal{I}_1)), \text{R}(\text{last}(\mathcal{I}_1))]$. It is easy to see that the later interval is the subset of ϕ_2 since $\text{R}(\text{last}(\mathcal{I}_1))$ appears before $\text{R}(\text{first}(\mathcal{I}_2))$ in the ordering of variables because of the definition of $\mathcal{I}_1$ and $\mathcal{I}_2$. Also because of the assumption that $\mathcal{I}$ is intersecting, $\text{L}(\text{last}(\mathcal{I}_2))$ appears before $\text{R}(\text{first}(\mathcal{I}_1))$ in the ordering. So the maximum readability of the formula generated by REDUCE2($\mathcal{I}$) where $\mathcal{I}$ consists of m terms satisfies $r_2(m) \leq 2 + r_2(m/2)$. Solving the recurrences yields the stated bound on the readability of $\mathcal{I}$. □

3.2 (p, q)-intersecting DNF

A monotone Boolean DNF is called (p, q)-*intersecting* if every p of its distinct terms intersect in at most q variables. A quadratic DNF for instance is $(2, 1)$-intersecting and k-DNF, i. e., DNF where the size of each term is bounded by k is $(2, k-1)$-intersecting. In this section, we give a $(p+q-1)m^{1-\frac{1}{q+1}}$ bound on the readability of (p, q)-intersecting DNF containing m-terms. Theorem 2 implies that this bound is almost tight because by considering $q+1$-homogeneous DNF containing $m = \Theta(n^{q+1})$ terms we get,

Corollary 1. *For a constant q, let $\mathscr{G}_q$ be the class of monotone Boolean DNF on n variables with m terms such that size of every minterm is exactly $q+1$. The readability of almost all $g \in \mathscr{G}_q$ is $\Omega(m^{1-\frac{1}{q+1}} \log^{-1} n)$.*

Procedure REDUCE3(ϕ, p, q)**:**

Input: A monotone Boolean (p, q)-intersecting DNF ϕ on variables set V

Output: A $(p+q-1)m^{1-\frac{1}{q+1}}$ readable formula ψ equivalent to ϕ

1. $\psi := 0,\ m := |\phi|$
2. **while** $\exists x \in V$ s.t. $\deg_\phi(x) \geq m^{1-\frac{1}{q+1}}$
3.   let $\phi_x = \bigvee_{t \in \phi, x \in t} t$
4. $\phi := \phi \setminus \phi_x$
5.   **if** $q > 1$ **then**
6. $\psi := \psi \vee (x \wedge \mathrm{REDUCE3}(\phi_x\ |_{x=1}, p, q-1))$
7.   **else**
8. $\psi := \psi \vee (x \wedge (\phi_x\ |_{x=1}))$
9. **return** $\phi \vee \psi$

Fig. 2. An algorithm to find $(p+q-1)m^{1-\frac{1}{q+1}}$ readable formula for (p, q)-intersecting DNF consists of m terms

Let ϕ be a (p, q)-intersecting monotone Boolean DNF on variables $V = \{x_1, \ldots, x_n\}$. The algorithm is given in Figure 2. It works by picking a variable x with high degree in ϕ and recursively computing a formula equivalent to the part of ϕ where x occurs. The algorithm stops when every variable has low degree in the remaining expression. More precisely, for a variable $x \in V$, let ϕ_x be the DNF consisting of terms of ϕ which contain x, i.e. $\phi_x = \bigvee_{t \in \phi, x \in t} t$. Note that if ϕ is (p, q)-intersecting then $\phi_x\ |_{x=1}$ is $(p, q-1)$-intersecting DNF, so the algorithm recurs when $q > 1$ and otherwise it returns the read-$(p-1)$ formula $x \wedge (\phi_x\ |_{x=1})$. The next Theorem bounds the readability of the formula generated by the algorithm.

Theorem 4. *Given a monotone Boolean DNF μ which is (p, q)-intersecting for $p \geq 2, q \geq 1$. The formula $\mu' = REDUCE3(\mu, p, q)$ is $(p+q-1)m^{1-\frac{1}{q+1}}$ readable and it is equivalent to μ.*

Proof. The proof is by induction on q. When $q = 1$, the while loop in Step 2 ensure that every variable in ϕ has degree less then $\sqrt{m}$ after the loop ends. Moreover, a read-$(p-1)$ formula is added to ψ in each iteration of while loop. Since there are at most $\sqrt{m}$ iterations, the formula $\phi \vee \psi$ in Step 9 has readability at most $\sqrt{m} + (p-1)\sqrt{m}$.

Now assume that the claim is true for $(p, q-1)$ intersecting DNF, where $q \geq 2$. We prove it for (p, q)-intersecting DNF using similar arguments as in the previous paragraph. After the while loop ends, every variable in the remaining ϕ has degree less then $m^{1-\frac{1}{q+1}}$. Let $m_1, \ldots, m_d$ be number of terms removed from ϕ in each iteration of while loop, where d is the number of iterations. Note that d can be bounded from above by $m^{\frac{1}{q+1}}$ since each m_i is at least $m^{1-\frac{1}{q+1}}$. Now, denoting the readability of (p, q)-intersecting DNF on m terms by $r_{p,q}(m)$, we have

$$r_{p,q}(m) \le m^{1-\frac{1}{q+1}} + \sum_{i=1}^{d} r_{p,q-1}(m_i) \le \ m^{1-\frac{1}{q+1}} + \sum_{i=1}^{d} \left((p+q-2) m_i^{1-\frac{1}{q}} \right) \quad (1)$$

$$\le m^{1-\frac{1}{q+1}} + (p+q-2)d \left(\frac{\sum_{i=1}^{d} m_i}{d} \right)^{1-\frac{1}{q}} \le \ (p+q-1) m^{1-\frac{1}{q+1}}, \quad (2)$$

where we apply induction hypothesis to get Equation (1) and use Jensen's inequality to get Equation (2).

The correctness of the procedure is straightforward since the invariant that $\phi \vee \psi$ is equal to μ holds after completion of every iteration. □

Note that the algorithm produces a depth q formula. In the next section we will see that we can do much better in this regard for the a subclass of (p, q)-intersecting DNF, namely the class of DNF where the size of each term is bounded by a constant k.

3.3 *k*-DNF

A monotone Boolean DNF is called k-DNF if every term in it has size at most k. In this section, we give an algorithm to compute $2km^{1-1/k}$ readable formula of depth three and equivalent to given k-DNF. We need the following definitions.

A *sunflower* with p *petals* and a *core* Y is a collection of sets $S_1, \ldots, S_p$ such that $S_i \cap S_j = Y$ for all $i \neq j$ and none of the sets $S_i - Y$ is empty. We allow the core Y to be empty however, so every pairwise disjoint family of sets constitutes a sunflower.

Lemma 1 (Sunflower Lemma [ER60]). *Let $\mathcal{H} \subseteq 2^V$ be a hypergraph with $m = |\mathcal{H}|$ and size of each edge is bounded by k. If $m > k!p^k$ then $\mathcal{H}$ contains a sunflower with $p+1$ petals.*

Since a sunflower has a straightforward read-once representation, the above lemma immediately gives an upper bound on the readability of k-DNF with m terms. The algorithm works by finding a sunflower with certain minimal size, representing them as read-once formula and recurse on the remaining edges.

Theorem 5. *Let f be a monotone Boolean DNF with m terms such that the size of each term in f is bounded by k then f is $2km^{1-1/k}$-readable. Moreover, a formula of such readability and depth 3 can be found in polynomial time.*

Proof. Any k-DNF with m terms contains a sunflower of size at least $(m/k!)^{1/k}$ which we remove and recurse on the remaining terms. Let $r(m)$ denote the readability of boolean k-DNF with m terms then the readability of f can be bounded by the recurrence $r(m) \le 1 + r(m - (m/k!)^{1/k})$ with $r(2) = r(1) = 1$. By using the inequality $k! \le k^k$ and substituting $r(m) = 2km^{1-1/k}$ in the above recurrence we get $g(k, m) = 2km^{1-\frac{1}{k}} \left(1 - \left(1 - \frac{m^{\frac{1}{k}-1}}{k} \right)^{1-\frac{1}{k}} \right) \ge 1$. Using elementary calculus, it can be proved that for $k \geq 2$ and $m \geq 1$, the function $g(k, m)$ is monotonically decreasing in m and monotonically increasing in k. Thus the minimum of g is attained when $k = 2$ and m approaches infinity. The minimum value is 1 and hence $r(m) \leq 2km^{1-1/k}$. Finally, we note that the proof of Lemma 1 is constructive and a sunflower of desired size can be computed in time polynomial in number of variables and terms of a DNF. $\square$

4 Hardness and Inapproximability

In this section, we show that finding the readability of a given monotone Boolean formula is NP-hard. The reduction we use is gap-introducing and so it also gives hardness of approximating readability unless P = NP. Our reduction is from the well-known NP-complete problem of deciding satisfiability of a given Boolean 3-CNF $\Phi(x_1 \dots x_n) = \bigwedge_{j=1}^{m} \Phi_j$. For all $i \in [n]$ and $j \in [m]$, let us define new variables y_{ij}, y'_{ij}, z'_{ij} for a literal x_i in clause Φ_j and variables z_{ij}, z'_{ij}, y'_{ij} for a literal $\neg x_i$ in clause Φ_j. Let $\phi(y_{11} \dots y_{nm}, z_{11} \dots z_{nm})$ be the monotone CNF we get from $\Phi(x_1 \dots x_n)$ by substituting y_{ij} for x_i in Φ_j and $z_{i'j}$ for $\neg x_{i'}$ in Φ_j such that $\phi(y, z) \equiv \Phi(x)$, for $y_{ij} = x_i$ and $z_{ij} = \neg x_i,\ i \in [n], j \in [m]$. Furthermore, let $I_i = \{j : x_i \in \Phi_j\} \cup \{j : \neg x_i \in \Phi_j\}$, we define

$$\rho(y', z') = \bigwedge_{i=1}^{n} \left(\bigwedge_{j \in I_i} y'_{ij} \vee \bigwedge_{j \in I_i} z'_{ij} \right), \quad \psi(y, z, y', z') = \bigvee_{x_i \in \Phi_j} y_{ij} z'_{ij} \vee \bigvee_{\neg x_i \in \Phi_j} y'_{ij} z_{ij}.$$

Now consider the following Boolean function

$$f(y, z, y', z') = \Big(\phi(y, z) \bigwedge \rho(y', z')\Big) \bigvee \psi(y, z, y', z'). \tag{3}$$

Note that the size of f is $15m$, where m is number of clauses in Φ. The next lemma shows that finding the readability of Boolean formula f defined in Equation (3) is equivalent to solving satisfiability for $\Phi(x)$.

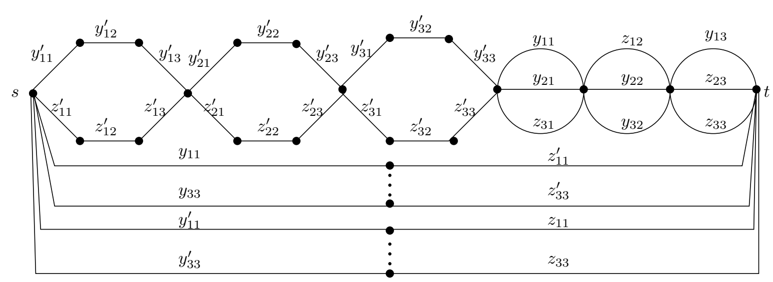

Fig. 3. Applying reduction in Equation (3) to 3-CNF $\Phi = (x_1 \vee x_2 \vee \neg x_3)(\neg x_1 \vee x_2 \vee x_3)(x_1 \vee \neg x_2 \vee \neg x_3)$. Minimal $s - t$ paths in the figure correspond to minterms of f, whereas minimal $s - t$ cuts are maxterms of f.

Lemma 2. *The monotone Boolean function f in Equation* (3) *is read-*2 *if and only if $\Phi(x)$ is satisfiable. It is read-once otherwise.*

Proof. Denote the two disjuncts in f by $f_1 = \phi(y,z) \wedge \rho(y',z')$ and $f_2 = \psi(y,z,y',z')$. We first show that the minterms of f_1 which are not absorbed by minterms of f_2 correspond precisely to the satisfiable assignments of Φ and so $f = \psi$ is clearly a read-once function if Φ is not satisfiable.

Let $\hat{x}$ be a satisfiable assignment of $\Phi(x)$. Since $\hat{x}$ makes at least one literal true in each clause of $\Phi(x)$, the set $t_\phi = \{y_{ij}|\hat{x}_i = 1\} \cup \{z_{ij}|\hat{x}_i = 0\}$ contains a minterm t'_ϕ of $\phi(y,z)$. Similarly, note that the set $t_\rho = \{y'_{ij}|\hat{x}_i = 1\} \cup \{z'_{ij}|\hat{x}_i = 0\}$ defines a minterm of $\rho(y',z')$, and so the set $t = t'_\phi \cup t_\rho$ is a minterm of f_1. It is easy to check that t does not contain any minterm of f_2 since for all $i \in [n]$ and $j \in [m]$, atmost one from each pair y_{ij}, z'_{ij} and y'_{ij}, z_{ij} are members of t.

Conversely, any minterm t of f_1 contains one of $y'_{i1} \dots y'_{im}$ or $z'_{i1} \dots z'_{im}$ for all $i \in [n]$ to cover the conjunct ρ. Assume t is not absorbed by any term of f_2. Consequently, t does not contain both $y_{ij}z'_{ij}$ or $y'_{ij}z_{ij}$ for all $i \in [n]$ and $j \in [m]$. Therefore it must contain from each clause ϕ_j, at least one of the variable y_{ij} or z_{ij} consistent with the primed variable selected from ρ. Hence the assignment $x_i = 1$ if $y_{ij} \in t$ and $x_i = 0$ if $z_{ij} \in t$ satisfies $\Phi(x)$.

It only remains to prove that f is not a read-once function when $\Phi(x)$ is satisfiable. Assume without loss of generality that the variable x_1 appears in clause Φ_1. Let us define a maxterm c of f by $c = \{y'_{11}, z'_{11}\} \bigcup_{i \in [n], j \in [m]} \{y_{ij}, y'_{ij}\}$ and consider the minterm t of f corresponding to a satisfiable assignment $\hat{x}$ of Φ as defined above. It is easy to see that $|t \cap c| > 1$ since for any literal x_i appears in clause Φ_j such that $\hat{x}_i = 1$, t would contain both y_{ij} and y'_{ij}. Hence f is not a read-once function because of Theorem 1. Note that it is read-2 since we have Equation (3) as its read-2 representation. □

Since f in Equation (3) is compose of two read-once formulae, Lemma 2 also implies the hardness of determining if a given monotone formula is disjunction of two read-once formulae .

Corollary 2. *It is* NP*-hard to know whether a given monotone Boolean formula is a read-once function or a disjunction of two monotone read-once functions.*

Another interesting problem for which we get a hardness result as a corollary of Lemma 2 is the problem of generating all minterms or maxterms of given monotone Boolean formula. Note that the problem can be solved in polynomial time [GG09] when the input formula is read-once.

Lemma 3. *Let $\mathscr{F}$ be the class of monotone Boolean formulae in which each variable appears at most twice. For a formula $f \in \mathscr{F}$, let C and D denote the sets of the maxterms and the minterms of f, respectively.*

(i) Given a formula $f \in \mathscr{F}$ and a subset C' of C, it is coNP*-complete to decide whether $C' = C$.*

(ii) Similarly, for a formula $f \in \mathscr{F}$ and a subset D' of D, it is coNP*-complete to decide whether $D' = D$.*

Proof. Note that since the class $\mathscr{F}$ is closed under duality, both parts of the theorem are equivalent. The hardness of (ii) implied immediately from Lemma 2 by setting $D' = \{t | t \text{ is a term in } \psi\}$. The (possibly) remaining minterms in $D \setminus D'$ correspond to satisfiable assignments of Φ. □

In the following, we generalize the reduction introduced in Equation (3) and get an inapproximability result for the problem of determining readability of given monotone Boolean formula. We use a result of Gál [Gό2] that gives an explicit monotone Boolean function α on s variables such that the size of the shortest monotone formula representing α is $s^{\Omega(\log s)}$, moreover its irredundant monotone DNF has size $s^{\mathcal{O}(\log s)}$. Note that the readability of α is also $s^{\Omega(\log s)}$, since otherwise we could represent α by a formula with smaller then shortest possible size. We define the following reduction

$$f'(w, y, z, y', z') = \Big(\phi(y, z) \bigwedge \rho(y', z') \bigwedge \alpha(w)\Big) \bigvee \psi(y, z, y', z'),$$

where the size of f' is $15m + s^{\mathcal{O}(\log s)}$. Note that if Φ is satisfiable, f' has readability $s^{\Theta(\log s)}$ by applying the same reasoning as in Lemma 2. By choosing s and m such that $m = s^{c_1 \log s}$ and $m = c_2 n$ for a suitable constants c_1, c_2, we get the following.

Corollary 3. *There is no polynomial-time algorithm to approximate the readability of a given monotone Boolean formula f within factor of $\mathcal{O}(n)$, unless* $\mathsf{P} = \mathsf{NP}$.

References

[AHK93] Dana Angluin, Lisa Hellerstein, and Marek Karpinski. Learning read-once formulas with queries. *J. ACM*, 40(1):185–210, 1993.

[ER60] P. Erdös and R. Rado. Intersection theorems for systems of sets. *J. London Math. Soc.*, 35:85–90, 1960.

[Gό2] Anna Gál. A characterization of span program size and improved lower bounds for monotone span programs. *Comput. Complex.*, 10(4):277–296, 2002.

[GG09] Martin Charles Golumbic and Vladimir Gurvich. Read-once functions. In Yves Crama and Peter Ladislaw Hammer, editors, *Boolean Functions: Theory, Algorithms and Applications.* Cambridge University Press, 2009. (in press).

[GMR06] Martin Charles Golumbic, Aviad Mintz, and Udi Rotics. Factoring and recognition of read-once functions using cographs and normality and the readability of functions associated with partial k-trees. *Discrete Applied Mathematics*, 154(10):1465–1477, 2006.

[Gur77] Vladimir Gurvich. On repetition-free boolean functions. *Uspekhi Mat. Nauk. (Russian Math. Surveys)*, 32:183–184, 1977. (in Russian).

[KBEG07] Leonid Khachiyan, Endre Boros, Khaled M. Elbassioni, and Vladimir Gurvich. On the dualization of hypergraphs with bounded edge-intersections and other related classes of hypergraphs. *Theor. Comput. Sci.*, 382(2):139–150, 2007.

[Weg87] Ingo Wegener. *The complexity of Boolean functions.* John Wiley & Sons, Inc., New York, NY, USA, 1987.