1. Introduction.

Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone Boolean function of n variables:

$$x \geq x' \implies f(x) \geq f(x') \quad \text{for any} \quad x, x' \in \{0,1\}^n.$$

Denote by

$$c = \bigwedge_{I \in C} \bigvee_{i \in I} x_i \tag{1.1}$$

the irredundant conjunctive normal form (CNF) of f, where C is the set of the prime implicates $I \subseteq \{1, \ldots, n\}$ of f. Note that the anti-characteristic vector of any prime implicate $I \in C$ is a maximal false vector of f, and vice versa. Thus, there is a natural one-to-one correspondence $C \rightleftharpoons MAX\{x | f(x) = 0\}$.

Similarly, let

$$d = \bigvee_{J \in D} \bigwedge_{j \in J} x_j \tag{1.2}$$

be the irredundant disjunctive normal form (DNF) of the function f, where D is the set of the prime implicants $J \subseteq \{1, \ldots, n\}$ of f. The characteristic vector of any prime implicant $J \subseteq \{1, \ldots, n\}$ is a minimal true vector of f, which gives a bijection $D \rightleftharpoons MIN\{x | f(x) = 1\}$. By definition,

$$f(x) = c(x) = d(x) \quad \text{for all } x \in \{0,1\}^n. \tag{1.3}$$

In this paper, we investigate the complexity of generating the irredundant normal forms c and/or d for various input representations of f. Let $\{\cdot\}$ denote either C, or D, or the set $C \sqcup D$ of all $|C| + |D|$ prime implicates and implicants of f. We consider the following problems:

$Gen\{\cdot\}$: *Given a subset $S \subseteq \{\cdot\}$, either prove that $S = \{\cdot\}$, or find a new element in $\{\cdot\} \setminus S$.*

Section 2 deals with problem $Gen\{C \sqcup D\}$. In Theorem 1 we show that this problem can be solved in incremental quasi-polynomial time provided that $f(x)$ can be evaluated for any $x \in \{0,1\}^n$ in polynomial time. Specifically, given two subsets $C' \subseteq C$ and $D' \subseteq D$ of total size $m = |C'| + |D'| < |C| + |D|$, a new element in $(C \setminus C') \cup (D \setminus D')$ can be generated in time $O(n(t + n)) + m^{o(\log m)}$, where t is the complexity of evaluating $f(x)$ at a binary point x. Note that this result implies that the condition $(C', D') = (C, D)$ can also be checked in $O(n(t+n)) + m^{o(\log m)}$ time.

An important special case of Theorem 1 is for $D' = D$. In such a case, f is already represented by its irredundant DNF and consequently $f(x)$ can be evaluated in polynomial time. Next, computing the irredundant CNF for f is equivalent to computing the irredundant DNF for the dual function $f^d(x) \doteq \neg f(\neg x)$. This problem is known as *Dualization* or *Transversal Hypergraph* — see e.g. Bioch and Ibaraki (1993), Boros, Gurvich, and Hammer (1993), Crama (1987), Fredman and Khachiyan (1994), Eiter and Gottlob (1995), Johnson, Yannakakis, and Papadimitriou (1988), Peled and Simeone (1994). Theorem 1 thus implies that the dualization problem for monotone DNFs can be solved in incremental quasi-polynomial time (Fredman and Khachiyan (1994) — see Theorem 2 below.) In fact, Theorem 1 rests upon this result, and the polynomial-time solvability of the dualization problem would imply the solvability of problem $Gen\{C \sqcup D\}$ in incremental polynomial time (Bioch and Ibaraki (1993)).

Another straightforward consequence of Theorem 1 is as follows. Suppose that $f(x)$ can be evaluated for each $x \in \{0,1\}^n$ in quasi-polynomial time $2^{\text{polylog}(\cdot)}$, where $(\cdot)$ is the size of the input encoding of f and x. Then the set $C \sqcup D$ can be constructed in time bounded by a quasi-polynomial in the *total* input and output size. Theorem 3 in Section 2 shows that, even for the class of $\wedge, \vee$-formulae f of depth 2, it is unlikely that uniform sampling from within $C \sqcup D$ can be carried out in time bounded by a quasi-polynomial $2^{\text{polylog}(\cdot)}$ in the input size of f. Specifically, the existence of such a randomized algorithm would imply that any NP-complete problem can be solved in quasi-polynomial time by a randomized algorithm with arbitrarily small failure probability. Our arguments are similar to those used by Jerrum, Valiant, and Vazirani (1986) for the problem of uniformly generating cycles in a digraph.

Finally, in Section 3 we consider problems $Gen\{C\}$ and $Gen\{D\}$. In Theorems 4-7 we show that for some natural classes of polynomial-time computable monotone Boolean functions it is NP-hard to test either of the conditions $C' = C$ or $D = D'$. Modulo the standard bijections $C \rightleftharpoons MAX\{x | f(x) = 0\}$ and $D \rightleftharpoons MIN\{x | f(x) = 1\}$, our examples of such sets C (or D) are as follows:

- All prime implicates (or implicants) of a $\wedge, \vee$-formula of depth 3;

- All minimal subsets of relays connecting (or disconnecting) two terminals in a monotone relay circuit;
- All minimal winning sets of Player 1 (or 2) for a positional game form with perfect information;
- All maximal feasible (or minimal infeasible) subsystems of a system of convex inequalities.

For each of the above examples, problems $Gen\{C\}$ and $Gen\{D\}$ cannot be solved in total (and hence incremental) quasi-polynomial time, unless any problem in NP is solvable in quasi-polynomial time. But for all these examples, Theorem 1 guarantees that problem $Gen\{C \sqcup D\}$ can be solved in incremental quasi-polynomial time.

2. Simultaneously Generating C and D.

In this section we show that problem $Gen\{C \sqcup D\}$ can be solved in incremental quasi-polynomial time.

Theorem 1. *Let $f : \{0,1\}^n \to \{0,1\}$ be a monotone Boolean function whose value at any point $x \in \{0,1\}^n$ can be determined in time t, and let C and D be the sets of the prime implicates and the prime implicants of f, respectively. Given two subsets $C' \subseteq C$ and $D' \subseteq D$ of total size $m = |C'| + |D'| < |C| + |D|$, a new element in $(C \setminus C') \cup (D \setminus D')$ can be found in time $O\left(n(t+n)\right) + m^{o(\log m)}$.*

As mentioned in the Introduction, Theorem 1 follows from its special case which deals with the dualization problem for monotone DNFs — cf. Bioch and Ibaraki (1993). For this reason, we start with the dualization problem:

Problem $(\mathcal{DD}^*)$: Given a pair of irredundant DNFs

$$d[A] = \bigvee_{I \in A} \bigwedge_{i \in I} x_i, \qquad d[B] = \bigvee_{J \in B} \bigwedge_{j \in J} x_j,$$

test whether $d[A]$ and $d[B]$ are mutually dual:

$$d[A](x_1, \ldots, x_n) = \neg d[B](\neg x_1, \ldots, \neg x_n) \quad \text{for all } x = (x_1, \ldots, x_n) \in \{0,1\}^n. \qquad (\mathcal{D})$$

If $d[A]$ and $d[B]$ are not dual, find a Boolean vector $x^* \in \{0,1\}^n$ such that

$$d[A](\neg x_1^*, \ldots, \neg x_n^*) = d[B](x_1^*, \ldots,\ x_n^*). \qquad (\mathcal{D}^*)$$

It is easy to see that any dual disjunctive normal forms $d[A]$ and $d[B]$ must satisfy the condition

$$I \cap J \neq \emptyset \text{ for all } I \in A \text{ and } J \in B. \tag{2.1}$$

For suppose to the contrary that there is a pair of disjoint sets $I \in A$ and $J \in B$. Then the characteristic vector of J satisfies $(\mathcal{D}^*)$.

Lemma 1 below shows that for any pair of dual irredundant forms we also have

$$\max\{|I| : I \in A\} \leq |B|, \quad \max\{|J| : J \in B\} \leq |A|. \tag{2.2}$$

Lemma 1. *Suppose that irredundant DNFs $d[A]$ and $d[B]$ satisfy (2.1). If condition (2.2) is violated, equation $(\mathcal{D}^*)$ can be solved in $O(|A| + |B| + n^2)$ time.*

Proof. First of all, (2.2) can be checked in $O(|A| + |B|)$ time. If $|J| > |A|$ for some $J \in B$, a solution x^* of equation $(\mathcal{D}^*)$ can be found as follows:

Initialize $x^* \leftarrow 0$

For each $I \in A$, select an index $i \in I \cap J$ and set $x_i^* \leftarrow 1$.

This procedure takes $O(n|A|)$ time. Since $|A| < |J| \leq n$, we obtain the time bound as required. Similarly, if $|I| > |B|$ for some $I \in A$, equation $(\mathcal{D}^*)$ can be solved in $O(n|B|)$ time. Again, $|B| < |I| \leq n$, which proves the lemma. □

Theorem 2. (Fredman and Khachiyan, 1994.) *Suppose that $d[A]$ and $d[B]$ satisfy (2.1) and (2.2). Then problem $(\mathcal{DD}^*)$ can be solved in time $v^{\chi(v)+O(1)}$, where $v = |A||B|$ and $\chi^\chi = v$.* □

From

$$\chi(v) \sim \log v \,/\, \log\log v = o(\log v),$$

the trivial inequality

$$v = |A||B| \leq (|A| + |B|)^2,$$

and Lemma 1 we obtain the following complexity bound.

Corollary 1. *If $d[A]$ and $d[B]$ satisfy (2.1), problem $(\mathcal{DD}^*)$ can be solved in time $T_{dual} = O(n^2) + (|A| + |B|)^{o(\log(|A|+|B|)}$.* □

Proof of Theorem 1. Suppose that $C' \subseteq C$ and $D' \subseteq D$, where C and D are defined by (1.1) and (1.2). For $A \subseteq C$, let $c[A] = \wedge_{I \in A} \vee_{i \in I} x_i$. With this notation, (1.3) implies

$c[C'](x) \geq c[C](x) \equiv c(x) \equiv f(x) \equiv d(x) \equiv d[D](x) \geq d[D'](x)$. Hence $(C', D') = (C, D)$ if and only if $c[C'](x) \equiv d[D'](x)$, which is equivalent to the duality of $d[C']$ and $d[D']$. In particular, we have $I \cap J \neq \emptyset$ for all $I \in C$ and $J \in D$. By Corollary 1, the duality of $d[C']$ and $d[D']$ can be tested in time $T_{dual} = O(n^2) + m^{o(\log m)}$, where $m = |C'| + |D'|$. If $(C', D') = (C, D)$, we are done. Otherwise we obtain a solution x^* of equation $(\mathcal{D}^*)$. It is easy to see that $c[C'](x^*) = 1$ and $d[D'](x^*) = 0$. Now we compute $f(x^*)$ and split into two cases.

Case 1: $f(x^*) = 0$. By evaluating $f(\cdot)$ at $O(n)$ binary points, we can find a vector $y^* \in MAX\{x|f(x) = 0\}$ such that $x^* \leq y^*$. Since f is monotone, $0 = f(y^*) < 1 = c[C'](x^*) \leq c[C'](y^*)$. This means that $I = \{i \mid y_i^* = 0,\ i = 1, \ldots, n\} \in C \setminus C'$, i.e., we obtain a new prime implicate of f.

Case 2: $f(x^*) = 1$. Find a vector $y^* \in MIN\{x|f(x) = 1\}$ such that $y^* \leq x^*$. The set $J = \{j \mid y_j^* = 1,\ \ j = 1, \ldots, n\} \in D \setminus D'$ is a new prime implicant of f. □

In the remainder of this section we discuss the complexity of uniformly sampling from $C \sqcup D$. A randomized algorithm $\mathcal{R}$ is an *ε-uniform generator* for a finite set Ω if

(i) $\mathcal{R}$ outputs only elements $\omega \in \Omega$, unless it stops with no output;

(ii) $\sum\{p(\omega)|\omega \in \Omega\} \geq 1/2$, where $p(\omega)$ is the probability that $\mathcal{R}$ outputs $\omega \in \Omega$;

(iii) $\max\{\ p(\omega)/p(\omega') \mid \omega, \omega'\} \leq 1 + \varepsilon$.

Theorem 3 below shows that a fast uniform generator for $C \sqcup D$ is unlikely to exist, even if we restrict the input to the class $\mathcal{DNF}_2$ of quadratic monotone DNFs. Note that the input size of any formula $f(x_1, \ldots, x_n) \in \mathcal{DNF}_2$ is polynomial in n.

Theorem 3. *Let $\rho < 1$ be a fixed constant, and let $\varepsilon = 2^{n^\rho}$. Suppose there exists a (quasi) polynomial-time randomized algorithm that, given a formula $f(x_1, \ldots, x_n) \in \mathcal{DNF}_2$, acts as an ε-uniform generator for the set $C \sqcup D$ of the prime implicates and implicants of f. Then any NP-complete problem can be solved in (quasi) polynomial time by a randomized algorithm with arbitrarily small one-sided failure probability.*

Proof of Theorem 3. Since for any formula $f \in \mathcal{DNF}_2$ the set D is given explicitly and $|D| \leq n^2$, any ε-uniform generator $\mathcal{R}$ for $C \sqcup D$ can be used as an ε-uniform generator for C. This entails at most $O(n^2)$ slowdown in the running time of $\mathcal{R}$. We can thus assume that there exists a (quasi) polynomial-time 2^{n^ρ}-uniform generator for C or equivalently, $MAX\{x|f(x) = 0\}$.

For a given graph $G = (V, E)$ with n vertices, define $f_G(x_1, \dots, x_n) = \vee\{x_i x_j \mid (ij) \notin E\}$. Then $MAX\{x | f_G(x) = 0\}$ is the set of (the characteristic vectors of) all maximal cliques in G. In other words, $\mathcal{R}$ can be used to 2^{n^ρ}-uniformly generate maximal cliques in G. To show that this implies the theorem, we need only slightly modify the proof suggested by Jerrum, Valiant and Vazirani (1986) for the problem of generating cycles in digraphs.

Let $H_k = \mathcal{K}_{2,2,\dots,2}$ be the complete k-partite graph, each "part" of which consists of two isolated vertices. Thus, H_k has $2k$ vertices and 2^k maximal cliques of size k each. Let $G(H_k)$ be the $2nk$-vertex graph obtained by substituting H_k for each vertex of G. Then $N(G(H_k), kl) = 2^{kl} N(G, l)$, where $N(\cdot, t)$ is the number of maximal cliques of size t in $(\cdot)$. Furthermore, $N(G(H_k), t) = 0$ if $t \neq 0 \ mod\ k$. Since the total number of cliques in G is bounded by 2^n, we conclude that for $k \geq 1 + n + (2nk)^\rho$, any $2^{(2nk)^\rho}$-uniform generator $\mathcal{R}$ of maximal cliques in $G(H_k)$ produces a clique of *maximum size* with probability $\geq 1/4$. Letting $k = \Theta(n^{1/(1-\rho)})$, we can satisfy the inequality $k \geq 1 + n + (2nk)^\rho$ and find a maximum clique in $G(H_k)$ with high probability in (quasi) polynomial time. But this is equivalent to solving the NP-complete clique problem for any input graph G. $\square$

The proof of Theorem 3 also shows that there is little hope that false vectors of a monotone quadratic DNF can be uniformly generated in polynomial time. It should be pointed out that Karp and Luby (1983) gave a simple polynomial-time algorithm for uniformly generating true vectors of an arbitrary, not necessarily monotone or quadratic, DNF.

We also mention in passing that problem $Gen\{x | x$ a maximal clique in $G\}$ can be solved in incremental polynomial time. In fact, all maximal cliques in a graph can be generated with polynomial delay — see Johnson, Yannakakis, and Papadimitriou (1988).

3. Generating C or D.

In this section we describe some classes of monotone Boolean functions for which it is NP-hard to separately check either of the the conditions $C' = C$ or $D' = D'$. This provides evidence that for each of these classes, problems $Gen\{C\}$ and $Gen\{D\}$ cannot be solved in total (or incremental) quasi-polynomial time. Our first example is as follows.

1. *Monotone Boolean Formulae of Depth 3.*

Theorem 4. *Let $\mathcal{F}_3$ be the class of $\wedge, \vee$-formulae of depth 3. For a formula $f \in \mathcal{F}_3$, let C and D denote the sets of the prime implicates and the prime implicants of f, respectively.*

(i) *Given a formula* $f \in \mathcal{F}_3$ *and a subset* C' *of* C*, it is coNP-complete to decide whether* $C' = C$*.*

(ii) *Similarly, for a formula* $f \in \mathcal{F}_3$ *and a subset* D' *of* D*, it is coNP-complete to determine whether* $D' = D$*.*

Proof. Since the class $\mathcal{F}_3$ is self-dual, parts (i) and (ii) of the theorem are equivalent. To show part (ii), it is convenient to state (ii) in the following equivalent form:

$\mathcal{E}$: *Given a formula* $f(x) \in \mathcal{F}_3$ *and a monotone DNF* $d(x)$ *such that* $f(x) \geq d(x)$ *for all* $x \in \{0,1\}^n$*, it is coNP-complete to check whether* $f(x) \equiv d(x)$*.*

It is well known that it is *coNP*-complete to test whether a given (non-monotone) DNF $D(x_1, \ldots, x_n)$ is a tautology. Substituting y_i for $\neg x_i$, $i = 1, \ldots, n$, we can transform $D(x_1, \ldots, x_n)$ into a monotone form $d(x_1, y_1, \ldots, x_n, y_n)$ such that

$$d(x,y) \equiv D(x) \quad \text{for} \quad y = \neg x.$$

Let $\phi(x,y) = \bigwedge_{i=1}^n (x_i \vee y_i)$. It is easy to see that $D(x)$ is a tautology, i.e.,

$$D(x) = 1 \quad \text{for all } x \in \{0,1\}^n,$$

if and only if

$$d(x,y) \vee \phi(x,y) = d(x,y) \quad \text{for all } x, y \in \{0,1\}^n.$$

Since $f(x,y) \doteq \phi(x,y) \vee d(x,y)$ is a $\wedge, \vee$-formula of depth 3 such that $f(x,y) \geq d(x,y)$, claim $\mathcal{E}$ and the theorem follow. □

Note that since any Boolean formula can be evaluated at any binary point in polynomial time, from Theorem 1 it follows that problem $Gen\{C \sqcup D\}$ can be solved in incremental quasi-polynomial time for any $\wedge, \vee$-formula. Observe also that any $\wedge, \vee$-formula of depth 2 is in conjunctive or disjunctive normal form. Theorem 2 thus implies that for $\wedge, \vee$-formulae of depth 2, problems $Gen\{C\}$ and $Gen\{D\}$ can be solved in incremental quasi-polynomial time. In addition, it is not hard to show that problems $Gen\{C\}$ and $Gen\{D\}$ can be solved with polynomial delay for any *read-once* $\wedge, \vee$-formula. (A formula is read-once if each variable appears in it exactly once — see e.g., Gurvich (1977) and Karchmer et al. (1993))

2. *Monotone Relay Circuits.*

Let $G = (V, E)$ be a graph with two distinguished vertices $s, t \in V$. A *monotone relay circuit* is a mapping $R : E \to \{1, \ldots, n\}$, which assigns a relay $R(e) \in \{1, \ldots, n\}$ to each edge $e \in E$ — cf. Shannon (1938). For a relay set $X \subseteq \{1, \dots, n\}$, let $ON(X) = \{e \in E \mid R(e) \in X\}$ and $OFF(X) = E \setminus ON(X)$. We say that X *connects* the terminals s and t if the graph $(V, ON(X))$ contains an s,t-path. Similarly, X *disconnects* the terminals if s and t are not connected in $(V, OFF(X))$. We shall call a minimal X connecting (disconnecting) the terminals s and t a *relay path* (*cut*), respectively.

Theorem 5. *Let Π be the class of series-parallel monotone relay circuits. Given a circuit in Π and a collection of relay s,t-cuts (or relay s,t-paths), it is coNP-complete to determine whether the given collection is complete.*

Proof. For a relay circuit R, let $f_R : \{0,1\}^n \to \{0,1\}$ be the monotone Boolean function realized by the circuit:

$$f_R(x_1, \dots, x_n) = \begin{cases} 1, & \text{if the set } \{i \mid x_i = 1,\ i = 1, \dots, n\} \text{ connects } s \text{ and } t; \\ 0, & \text{otherwise.} \end{cases} \tag{3.1}$$

Clearly, each relay s,t-cut (path) is a prime implicate (implicant) of f_R, and vice versa. Since any $\wedge, \vee$-formula can be easily realized by a series-parallel relay circuit, Theorem 5 follows from Theorem 4. □

As before, $f_R(x)$ can be evaluated for each binary vector x in polynomial time. Hence all relay cuts and paths in an arbitrary monotone relay circuit can be jointly generated in incremental quasi-polynomial time.

If the relay mapping $R : E \to \{1, \dots, n\}$ is bijective, the relay cuts and paths turn into the usual cuts and paths, which can be (separately) generated with polynomial delay for any graph G.

3. *Positional Two-Person Games with Perfect Information.*

Let $G = < V, E >$ be a directed acyclic graph with a distinguished vertex s such that all vertices $v \in V$ are reachable from s. A *two-person positional game on* G is a partitioning

$$V = V_1 \cup V_2, \quad V_1 \cap V_2 = \emptyset, \tag{3.2}$$

where V_1 and V_2 are the sets of positions controlled by Players 1 and 2, respectively.

Let $E^+(v)$ be the set of arcs incident from a position $v \in V$. The game starts in the initial position s. If the current position v is in V_α, $\alpha = 1, 2$, Player α selects a move from $E^+(v)$

until the game reaches a final position $u \in V_T = \{v \in V \mid E^+(v) = \emptyset\}$. The player who controls the final position wins — cf. Zermelo (1913).

A *game form* Γ specifies the partitioning (3.2) on $V \setminus V_T$, but does not indicate the winners on the set V_T of final positions. A subset $X \subseteq V_T$ is called a *winning set of Player* α if this player can force the game to finish in X, regardless of the adversary's moves.

Theorem 6. *Let $\alpha = 1$ or 2. Given a positional game form Γ and a list of minimal winning sets of Player α, it is coNP-complete to decide whether the given list is exhaustive.*

Proof. Assume $V_T = \{1, \ldots, n\}$ and consider the following Boolean function:

$$f_\Gamma(x_1, \ldots, x_n) = \begin{cases} 1, & \text{if the set } \{i \mid x_i = 1,\ i = 1, \ldots, n\} \text{ is a winning set of Player 1;} \\ 0, & \text{otherwise} \end{cases}$$

— see Gurvich (1973, 1975). Clearly f_Γ is monotone, and each minimal winning set of Player 1 is a prime implicant of f_Γ and vice versa. Furthermore, the prime implicates of f_Γ are nothing but the minimal winning sets of Player 2. It is also easy to see that any $\wedge, \vee$-formula of size l and depth d can be realized by a game form of the same size and depth. For this reason, Theorem 6 follows from Theorem 4. $\square$

Any positional game with perfect information can be solved in polynomial time by dynamic programming. Hence $f_\Gamma(x)$ can be evaluated for each x in polynomial time. From Theorem 1 we conclude that all minimal winning sets of Players 1 and 2 can be jointly generated in incremental quasi-polynomial time.

Let us remark that due to the obvious one-to-one correspondence between positional game forms and combinatorial $\wedge, \vee$-circuits, all minimal winning sets of each player can be generated with polynomial delay for positional games on trees.

4. *Convex Programming.*

Let $\mathcal{P} = (P_1, \ldots, P_n)$ be a system of polyhedra in $\Re^d$, and consider the monotone Boolean function

$$f_{\mathcal{P}}(x_1, \ldots, x_n) = \begin{cases} 1, & \text{if } \bigcap_{\{i \mid x_i = 1\}} P_i = \emptyset; \\ 0, & \text{otherwise.} \end{cases} \tag{3.3}$$

By definition, each maximal false vector of $f_{\mathcal{P}}$ corresponds to a maximal feasible subsystem of polyhedra from $\mathcal{P}$, whereas each minimal true vector of $f_{\mathcal{P}}$ can be viewed as a minimal infeasible subsystem of $\mathcal{P}$.

As an example, suppose that $\mathcal{P}$ is the set of all facets of a polytope $Q = \{y \in \Re^d \mid a_i y \leq b_i,\ i = 1, \ldots, n\}$. Then problem $Gen\{x | x$ a maximal feasible subsystem of $\mathcal{P}\}$ is equivalent to generating all vertices of Q, and can be solved in incremental polynomial time — see e.g. Chvátal (1983). In general, however, generating all maximal feasible subsystems of a system of polyhedra is hard. Analogously, generating all minimal infeasible systems of $\mathcal{P}$ can also be hard:

Theorem 7. *For a system $\mathcal{P}$ of nonempty polyhedra in $\Re^d$ and a collection of maximal feasible (minimal infeasible) subsystems of $\mathcal{P}$, it is coNP-complete to tell whether the given collection is complete.*

Proof. Let $R : E \to \{1, \ldots, n\}$ be an arbitrary relay circuit on a series-parallel graph $G = (V, E)$. It is easy to see that for any edge $e \in E$, all s-t-paths through e cross e in the same direction, which we refer-to as the s-t-orientation of e.

Denote by G_i the graph $(V, OFF(\{x_i\}))$ with the s-t-orientation on the set of its edges, and let P_i be the s-t-flow polyhedron for the digraph G_i, $i = 1, \ldots, n$. In other words, P_i consists of all vectors $y \in \Re^E$ such that

$$
\begin{aligned}
& y(e) = 0, \quad e \in ON(\{x_i\}), \\
& y(e) \geq 0, \quad e \in OFF(\{x_i\}), \\
& \sum \{\ y(e) \mid e \text{ incident from } s\ \} = 1, \\
& \sum \{\ y(e) \mid e \text{ incident from } v\ \} - \sum \{\ y(e) \mid e \text{ incident to } v\ \} = 0, \quad v \in V \setminus \{s, t\}.
\end{aligned}
$$

For this polyhedral system we have $f_{\mathcal{P}}(x) \equiv \neg f_R(\neg x)$, i.e., definitions (3.3) and (3.1) give mutually dual Boolean functions. This means that Theorem 7 is a corollary of Theorem 5. $\square$

Since linear programming is polynomial-time solvable, from definition (3.3) it follows that $f_{\mathcal{P}}(x)$ can be computed for each $x \in \{0, 1\}^n$ in polynomial time. Again, we conclude that all maximal feasible and minimal infeasible subsystems of an arbitrary system of convex polyhedral sets can be jointly generated in incremental quasi-polynomial time.

Acknowledgements: The first author gratefully acknowledge the partial support of the Office of Naval Research under Grants N0014-92-J-1375 and N0014-92-J-4083. The second author's research was supported in part by ONR Grant N0014-92-J-1375 and NSF Grants CCR-9208371 and CCR-9208539.

References

Bioch J. C. and T. Ibaraki (1993) Complexity of Identification and Dualization of Positive Boolean Functions, *RUTCOR Research Report 25-93*, Rutgers Center for Operations Research, Rutgers University, New Brunswick, New Jersey 08903.

Boros E., V. Gurvich, and P. L. Hammer (1993) Dual Subimplicants of Positive Boolean Functions, *RUTCOR Research Report 11-93*, Rutgers Center for Operations Research, Rutgers University, New Brunswick, New Jersey 08903.

Chvátal V. (1983) *Linear Programming*, Freeman, New York.

Crama Y. (1987) Dualization of Regular Boolean Functions, *Discrete Applied Mathematics* **16** 79-85.

Eiter T. and G. Gottlob (1995) Identifying the Minimal Transversals of a Hypergraph and Related Problems, to appear in SIAM J. Computing **24-6**.

Fredman M. and L. Khachiyan (1994) On the Complexity of Dualization of Monotone Disjunctive Normal Forms, *Technical Report LSCR-TR-225*, Department of Computer Science, Rutgers University, New Brunswick, New Jersey 08903.

Gurvich V. (1973) To Theory of Multistep Games, *USSR Comput. Math. and Math Phys.* **13-6** 1485-1500.

Gurvich V. (1975) Solvability of Positional Games in Pure Strategies, *USSR Comput. Math and Math. Phys.* **15-2** 357-371.

Gurvich V. (1977) On Repetition-Free Boolean Functions, *Uspekhi Mat. Nauk* **32** 183-184 (in Russian).

Jerrum M. R., L. G. Valiant, and V. V. Vazirani (1986) Random Generation of Combinatorial Structures From a Uniform Distribution, *Theoretical Computer Science* **43** 169-188.

Johnson D. S., M. Yannakakis, and C. H. Papadimitriou (1988) On Generating All Maximal Independent Sets, *Information Processing Letters* **27** 119-123.

Karchmer M., N. Linial, I. Newman, M. Saks, and A. Widgerson (1993) Combinatorial Characterization of Read-Once Formulae, *Discrete Mathematics* **114** 275-282.

Karp R. M. and M. Luby (1983) Monte-Carlo Algorithms for Enumeration and Reliability Problems, *Proc. 24th IEEE Symp. on Foundations of Computer Science* 56-64.

Peled U. N. and B. Simeone (1994) An $O(nm)-$Time Algorithm for Computing the Dual of a Regular Boolean Function, *Discrete Applied Mathematics* **49** 309-323.

Shannon C. E. (1938) A Symbolic Analysis of Relay and Switching Circuits, *Trans. AIEE* **57** 713-723.

Zermelo E. (1913) Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels, *Proc. 5th Congress of Mathematicians, Cambridge 1912,* Cambridge Univ. Press, 501-504.