Comput. complex. **7** (1998), 152 – 162
1016-3328/98/020152–11 \$ 1.50+0.20/0

computational complexity

SYMMETRIC ALTERNATION CAPTURES BPP

Alexander Russell and Ravi Sundaram

Abstract. We introduce the natural class $\mathbf{S}_2^P$ containing those languages that may be expressed in terms of two *symmetric quantifiers*. This class lies between Δ_2^P and $\Sigma_2^P \cap \Pi_2^P$ and naturally generates a "symmetric" hierarchy corresponding to the polynomial-time hierarchy. We demonstrate, using the probabilistic method, new containment theorems for **BPP**. We show that **MA** (and hence **BPP**) lies within $\mathbf{S}_2^P$, improving the constructions of Sipser and Lautemann which show that $\mathbf{BPP} \subseteq \Sigma_2^P \cap \Pi_2^P$. Symmetric alternation is shown to enjoy two strong structural properties which are used to prove the desired containment results. We offer some evidence that $\mathbf{S}_2^P \neq \Sigma_2^P \cap \Pi_2^P$ by constructing an oracle O such that $\mathbf{S}_2^{P,O} \neq \Sigma_2^{P,O} \cap \Pi_2^{P,O}$, assuming that the machines make only "positive" oracle queries.

Key words. Alternation, complexity classes, symmetric quantifiers, randomness
Subject classifications. 68Q15.

1. Introduction

Since the inclusion of randomness among those resources for which we have accepted computational models, determining the exact relationship between randomness and other such computational resources has become a major project. The relationship between space and randomness, elucidated by startling pseudorandom constructions (Ajtai *et al.* 1987, Nisan & Zuckerman 1993), is relatively well understood. These constructions demonstrate that space-bounded computations benefit little from the use of randomness. The analogous relationship with time has proved less tractable, the only (nontrivial) relationships depend on unproved complexity-theoretic assumptions (Impagliazzo *et al.* 1989, Yao 1982). We continue the study initiated by Sipser (1983) of the relationship between randomness and quantification, that is, the relationship between **BPP** and classes arising from appropriate quantification of polynomial-time predicates (e.g., **NP**, **coNP** and other classes in the polynomial-time hierarchy (Stockmeyer 1976)).

BPP was first shown to lie in the polynomial-time hierarchy by Sipser (1983) who demonstrated that $\mathbf{BPP} \subseteq \Sigma_2^P$ (and hence that $\mathbf{BPP} \subseteq \Sigma_2^P \cap \Pi_2^P$). We introduce a natural quantified class, $\mathbf{S}_2^P$, and demonstrate that

$$\mathbf{BPP} \subseteq \mathbf{S}_2^P \subseteq \Sigma_2^P \cap \Pi_2^P.$$

The class $\mathbf{S}_2^P$ consists of those languages $\mathcal{L}$ which may be decided by a polynomial-time machine that receives counsel from two provers in such a manner that when the input w is in $\mathcal{L}$, there is a witness x, which the first prover may provide, so that, regardless of the information provided by the second prover, the machine accepts. Similarly, when $w \notin \mathcal{L}$, there is a witness y, which the second prover may supply, so that, regardless of the information supplied by the first prover, the machine rejects. We refer to this special kind of alternation as *symmetric alternation*. The $\mathbf{S}_2$ operator, defined below, enjoys some remarkable structural properties:

- $\mathbf{S}_2 \cdot \mathbf{BP} \cdot \mathbf{P} \subseteq \mathbf{S}_2 \cdot \mathbf{P}$,
- $\mathbf{P}^{(\mathbf{S}_2 \cdot P)} = \mathbf{S}_2 \cdot P$.

We use these structural properties to conclude that

- $\mathbf{BPP} \subseteq \mathbf{MA} \subseteq \mathbf{S}_2^P$, and
- $\Delta_2^P = \mathbf{P}^{\mathbf{NP}} \subseteq \mathbf{S}_2^P$.

In light of the containment $\mathbf{BPP} \subseteq \mathbf{S}_2^P \subseteq \Sigma_2^P \cap \Pi_2^P$, the relationship between $\mathbf{S}_2^P$ and $\Sigma_2^P \cap \Pi_2^P$ is of natural interest. One standard method of offering evidence that two classes are different is to demonstrate an oracle which separates them. In Section 3, we construct an oracle which separates S_2^P from $\Sigma_2^P \cap \Pi_2^P$ under the assumption that the machines involved are monotone. The framework we develop to build this oracle can be used to simplify the construction given by Baker & Selman (1979) of an oracle separating Σ_2^P and Π_2^P.

2. Definitions and containment results

In what follows, Σ is used to denote the alphabet and may be assumed to be $\{0, 1\}$ without loss of generality. Throughout, the variable n denotes $|w|$, the length of the input in question. For $m \in \mathbb{N}$, we use $\exists^m x$ as shorthand for $\exists x(|x| = m)$, $\exists^m!x$, if such an x is unique ($\forall^m x$ and $\forall^m!x$ are similarly used).

Definition 2.1. *For a complexity class $\mathfrak{C}$, we define $\mathbf{S}_2 \cdot \mathfrak{C}$ to be the complexity class consisting of those languages $\mathcal{L}$ for which there exists a $\mathcal{C} \in \mathfrak{C}$ and a polynomial q such that*

- $w \in \mathcal{L} \Rightarrow \exists^{q(n)} x \ \forall^{q(n)} y \ \langle w, x, y \rangle \in \mathcal{C}$,
- $w \notin \mathcal{L} \Rightarrow \exists^{q(n)} y \ \forall^{q(n)} x \ \langle w, x, y \rangle \notin \mathcal{C}$.

Notice that the acceptance criterion for the $\mathbf{S}_2$ operator has the form of the acceptance criterion for Σ_2^P. The rejection criterion is similarly related to that of Π_2^P. The complexity class $\mathbf{S}_2 \cdot \mathbf{P}$, then, is clearly inside $\Sigma_2^P \cap \Pi_2^P$. Notice that $\mathbf{S}_2^P$, like **BPP** and **IP**, is a *promise* class—the criteria for acceptance and rejection are not complements of each other.

Definition 2.2. $\mathbf{S}_2^P \stackrel{\text{def}}{=} \mathbf{S}_2 \cdot \mathbf{P}$.

Definition 2.3. $\mathbf{S}_{2k}^P \stackrel{\text{def}}{=} \overbrace{\mathbf{S}_2 \cdot \mathbf{S}_2 \cdots \mathbf{S}_2}^{k} \cdot \mathbf{P}$.

Theorem 2.4. $\Sigma_1^P \bigcup \Pi_1^P \subseteq \mathbf{S}_2^P \subseteq \Sigma_2^P \bigcap \Pi_2^P$.

Proof. $\Sigma_1^P \subseteq \mathbf{S}_2^P \subseteq \Sigma_2^P$ and $\mathbf{S}_2^P$ is closed under complement. $\square$

Corollary 2.5. $\Sigma_k^P \bigcup \Pi_k^P \subseteq \mathbf{S}_{2k}^P \subseteq \Sigma_{2k}^P \bigcap \Pi_{2k}^P$.

Proof. By induction on k. $\square$

This allows us to conclude that $\mathbf{PH} \stackrel{\text{def}}{=} \bigcup_k \Sigma_2^P = \bigcup_k \mathbf{S}_{2k}^P$, so that the $\mathbf{S}_{2k}^P$, for $k \geq 1$, form a hierarchy which collapses if and only if the polynomial hierarchy collapses.

Theorem 2.6. $\mathbf{P}^{\mathbf{S}_2^P} \subseteq \mathbf{S}_2^P$.

Proof. Let $\mathfrak{S} \in \mathbf{S}_2^{\mathfrak{P}}$ and $S(\cdot,\cdot,\cdot)$ be a polynomial-time machine accepting $\mathfrak{S}$ according to the definition of $\mathbf{S}_2^P$. Let $\mathcal{L} \in \mathbf{P}^{\mathfrak{S}}$ and let $D^{\mathfrak{S}}$ be a deterministic polynomial-time machine deciding $\mathcal{L}$. A *computation suggestion* of $D^{\mathfrak{S}}(w)$ consists of a sequence of pairs $(\mathfrak{d}_i, \mathfrak{a}_i)$ for $1 \leq i \leq t$ such that

- each $\mathfrak{d}_i$ is an instantaneous description of $D^{\mathfrak{S}}$ (see Hopcroft & Ullman 1979 for a definition of instantaneous description);
- $\mathfrak{d}_1$ is the initial instantaneous description of $D^{\mathfrak{S}}(w)$;

- $\mathfrak{d}_t$ is a final instantaneous description of $D^{\mathfrak{S}}$;
- if $\mathfrak{d}_i$ does not find $D^{\mathfrak{S}}$ in its query state, then $\mathfrak{d}_i \vdash_D \mathfrak{d}_{i+1}$;
- if $\mathfrak{d}_i$ finds $D^{\mathfrak{S}}$ querying q_i, then there is a response r_i such that $\mathfrak{d}_i \vdash_D \mathfrak{d}_{i+1}$ with response r_i, and $\mathfrak{a}_i$ is a string of length appropriate for the quantified inputs of S on input q_i.

We construct a machine $T(\cdot,\cdot,\cdot)$ accepting $\mathcal{L}$ according to the definition of $\mathbf{S}_2^P$. $T(w,x,y)$ first examines x, rejecting unless x is a computation suggestion for $D^{\mathfrak{S}}$ such that $x = (\mathfrak{d}_i^x, \mathfrak{a}_i^x)_{i \le t_x}$. T then examines y, accepting unless y is a computation suggestion for $D^{\mathfrak{S}}$ such that $y = (\mathfrak{d}_i^y, \mathfrak{a}_i^y)_{i \le t_y}$. If $t_x = t_y$ and $\mathfrak{d}_i^x = \mathfrak{d}_i^y$ for all $1 \le i \le t_x$, then T accepts exactly when $\mathfrak{d}_{t_x}^x = \mathfrak{d}_{t_y}^y$ is an accepting description. Otherwise, there is an i_0 such that $\mathfrak{d}_{i_0}^x = \mathfrak{d}_{i_0}^y$ but $\mathfrak{d}_{i_0+1}^x \ne \mathfrak{d}_{i_0+1}^y$. Evidently, $\mathfrak{d}_{i_0}^x$ is a query state of $D^{\mathfrak{S}}$ (otherwise, there is a unique next state, upon which both x and y must agree if they are computation suggestions). Let q_{i_0} be the query appearing in this description. Assume without loss of generality that the computation suggestion of x claims that $q_{i_0} \in \mathfrak{S}$. Then, T simulates $S(q_{i_0}, \mathfrak{a}_{i_0}^x, \mathfrak{a}_{i_0}^y)$, accepting exactly when S accepts. Notice that for an input w, there is a *correct* computation suggestion $\Delta_w = (\mathfrak{d}_i, \mathfrak{a}_i)_{i \le t}$ (that is, one in which every oracle query is answered correctly) such that, for each query q_i,

- if $q_i \in \mathfrak{S}$, then $S(q_i, \mathfrak{a}_i, z)$ accepts for all z,
- if $q_i \notin \mathfrak{S}$, then $S(q_i, z, \mathfrak{a}_i)$ rejects for all z.

Hence,

- if $w \in \mathcal{L}$, then $T(w, \Delta_w, y)$ accepts for all $y \in \Sigma^{q(n)}$,
- if $w \notin \mathcal{L}$, then $T(w, x, \Delta_w)$ rejects for all $x \in \Sigma^{q(n)}$. □

COROLLARY 2.7. $\Delta_2^P = \mathbf{P}^{\mathbf{NP}} \subseteq \mathbf{P}^{\mathbf{S}_2^P} \subseteq \mathbf{S}_2^P$.

COROLLARY 2.8. $\Delta_{k+1}^P \subseteq \mathbf{S}_{2k}^P$.

The proof of Theorem 2.9 below is a generalization of the argument of Lautemann (1983).

THEOREM 2.9. $\mathbf{S}_2 \cdot \mathbf{BP} \cdot \mathbf{P} \subseteq \mathbf{S}_2 \cdot \mathbf{P}$.

PROOF. Let $\mathcal{L} \in \mathbf{S}_2 \cdot \mathbf{BP} \cdot \mathbf{P}$. Let $\mathcal{D} \in \mathbf{P}$ and let q, r be polynomials such that

- $w \in \mathcal{L} \implies \exists^{q(n)}x\ \forall^{q(n)}y\ \Pr_{r \in_R \Sigma^{r(n)}}[\langle w, x, y, r\rangle \in \mathcal{D}] \geq 1 - 2^{-q(n)-n}$,
- $w \notin \mathcal{L} \implies \exists^{q(n)}y\ \forall^{q(n)}x\ \Pr_{r \in_R \Sigma^{r(n)}}[\langle w, x, y, r\rangle \in \mathcal{D}] \leq 2^{-q(n)-n}$.

Fix $w \in \Sigma^*$, and let $\hat{x} \in \Sigma^{q(n)}$ be such that

$$\forall y \in \Sigma^{q(n)} \quad \Pr_{r \in_R \Sigma^{r(n)}}[\langle w, \hat{x}, y, r\rangle \in \mathcal{D}] \geq 1 - 2^{-q(n)-n}.$$

Let $\mathcal{W}_y \subseteq \Sigma^{r(n)}$ be the collection of random strings r for which $\langle w, \hat{x}, y, r\rangle \in \mathcal{D}$ and let $\mathcal{W} \stackrel{\text{def}}{=} \bigcap_y \mathcal{W}_y$. These are the random strings r such that $\langle w, \hat{x}, y, r\rangle \in \mathcal{D}$ for all y. For a set B, let $\mu(B)$ denote the measure of the set. Then, $\forall y\ \mu(\mathcal{W}_y) \geq 1 - 2^{-q(n)-n}$, so that $\mu(\mathcal{W}) \geq 1 - 2^{-n}$. We demonstrate that there exists a sequence $\sigma \stackrel{\text{def}}{=} (\sigma_1, \ldots, \sigma_{r(n)})$ of elements of $\Sigma^{r(n)}$, so that for every $\tau \in \Sigma^{r(n)}$, there is some i such that $\sigma_i \oplus \tau \in \mathcal{W}$ ($\oplus$ stands for the binary operator that returns the bitwise XOR of the operands). Selecting $\sigma_1, \ldots, \sigma_{r(n)}$ uniformly and independently at random from $\Sigma^{r(n)}$, let $\mathcal{B}_\tau$ be the event such that for each i, $\sigma_i \oplus \tau \notin \mathcal{W}$. Then, $\Pr[\mathcal{B}_\tau] \leq 2^{-nr(n)}$, so that

$$\Pr\left[\bigvee_{\tau \in \Sigma^{r(n)}} \mathcal{B}_\tau\right] \leq \sum_\tau \Pr[\mathcal{B}_\tau] = 2^{r(n)} 2^{-nr(n)} = 2^{r(n)(1-n)} < 1.$$

Hence, there is a sequence $\sigma = (\sigma_1, \ldots, \sigma_{r(n)})$, so that for all τ, there is an i such that $\sigma_i \oplus \tau \in \mathcal{W}$.

Suppose now that $w \notin \mathcal{L}$. Then, there is a $\hat{y} \in \Sigma^{q(n)}$ such that for all $x \in \Sigma^{q(n)}$, $\Pr_r[\langle w, x, \hat{y}, r\rangle \in \mathcal{D}] \leq 2^{-q(n)-n}$. As before, let

$$\mathcal{W}_x \stackrel{\text{def}}{=} \left\{r \in \Sigma^{r(n)} \mid \langle w, x, \hat{y}, r\rangle \in \mathcal{D}\right\}.$$

Define $\mathcal{W} \stackrel{\text{def}}{=} \bigcup_x \mathcal{W}_x$ so that $\mu(\mathcal{W}) \leq 2^{-n}$. Then, we show that there is a sequence $\tau \stackrel{\text{def}}{=} (\tau_1, \ldots, \tau_{r(n)^2})$ of elements of $\Sigma^{q(n)}$ such that for all $\sigma = (\sigma_1, \ldots, \sigma_{r(n)})$, $\exists j\ \forall i\ \sigma_i \oplus \tau_j \notin \mathcal{W}$. Selecting $\tau_1, \ldots, \tau_{r(n)^2}$ independently and uniformly at random, let $\mathcal{B}_\sigma$ be the event such that $\forall j\ \exists i\ \sigma_i \oplus \tau_j \in \mathcal{W}$. Then,

$$\Pr[\mathcal{B}_\sigma] \leq \prod_j \sum_{i=1}^{r(n)} \Pr[\sigma_i \oplus \tau_j \in \mathcal{W}] = \prod_j \sum_{i=1}^{r(n)} 2^{-n} = (\frac{r(n)}{2^n})^{r(n)^2}.$$

Hence,

$$\Pr\left[\bigvee_\sigma \mathcal{B}_\sigma\right] \leq \sum_\sigma \Pr[\mathcal{B}_\sigma] = 2^{r(n)^2} \left(\frac{r(n)}{2^n}\right)^{r(n)^2} < 1.$$

Therefore, there is a sequence $(\tau_1, \ldots, \tau_{r(n)^2})$ with the property that for any sequence $(\sigma_1, \ldots, \sigma_{r(n)})$, there is some j, such that $\sigma_i \oplus \tau_j \notin \mathcal{W}$ for all i. In light of this, consider the deterministic polynomial-time machine $M(\cdot, \cdot, \cdot)$ which, on input (w, α, β),

- checks the format of α, rejecting unless $\alpha = \langle x; \sigma_1, \ldots, \sigma_{r(n)} \rangle$,
- checks the format of β, accepting unless $\beta = \langle y; \tau_1, \ldots, \tau_{r(n)^2} \rangle$, and
- accepts iff for each τ_j, there is some σ_i such that $\langle w, x, y, \sigma_i \oplus \tau_j \rangle \in \mathcal{D}$.

From above, if $x \in \mathcal{L}$, then setting α to be the $\langle \hat{x}; \sigma_1, \ldots, \sigma_{r(n)} \rangle$ promised in the first part of the above discussion, we have that D accepts regardless of β. Similarly, if $x \notin \mathcal{L}$, then setting β to be the $\langle \hat{y}; \tau_1, \ldots, \tau_{r(n)^2} \rangle$ promised in the second part of the above discussion, we have that D rejects regardless of α. $\square$

Corollary 2.10. $\mathbf{MA} \subseteq \mathbf{S}_2^P$.

Proof. $\mathbf{MA} = \exists \cdot \mathbf{BP} \cdot \mathbf{P} \subseteq \mathbf{S}_2 \cdot \mathbf{BP} \cdot \mathbf{P} \subset \mathbf{S}_2 \cdot \mathbf{P}$. $\square$

Corollary 2.11. $\mathbf{BPP} \subseteq \mathbf{S}_2^P$.

3. An oracle separating monotone $\mathbf{S}_2^P$ and monotone $\mathbf{\Sigma}_2^P \cap \mathbf{\Pi}_2^P$

Definition 3.1. *We shall call an oracle Turing machine M^O* monotone, *if for any inputs $x_1, \ldots, x_n$ and oracles O_1, O_2 we have*

$$O_1 \subseteq O_2 \wedge M^{O_1}(x_1, \ldots, x_n) \text{ accepts} \implies M^{O_2}(x_1, \ldots, x_n) \text{ accepts.}$$

We then define $\mathbf{mS}_2^{P,O}$ to be the class of languages accepted by some monotone machine according to the $\mathbf{S}_2^P$ acceptance rules with oracle O. The classes $\mathbf{m}\Sigma_2^{P,O}$ and $\mathbf{m}\Pi_2^{P,O}$ are defined similarly.

The above conclusion that $\mathbf{BPP} \subseteq \mathbf{S}_2^P \subseteq \Sigma_2^P \cap \Pi_2^P$ is interesting only to the extent to which we believe that the inclusion $\mathbf{S}_2^P \subseteq \Sigma_2^P \cap \Pi_2^P$ is strict. We offer evidence for the strictness of this inclusion by constructing an oracle O such that $\mathbf{mS}_2^{P,O} \subsetneq \mathbf{m}\Sigma_2^{P,O} \cap \mathbf{m}\Pi_2^{P,O}$.

We begin with some definitions relevant to our construction. For $n \geq 0$, consider subsets T of Σ^{2n} satisfying the Π_2^P predicate $\forall^n x \; \exists^n y \; xy \in T$. This predicate is monotone. Collect together its minterms to form

$$\mathfrak{T} = \{T \subseteq \Sigma^{2n} \mid \forall^n x \; \exists^n ! y \; xy \in T\}.$$

This set has size 2^{n2^n}. Given a family of minterms $\mathcal{T} \subseteq \mathfrak{T}$ and a set $W \subseteq \Sigma^{2n}$, we define

$$\mathcal{T}_W \stackrel{\text{def}}{=} \{T \in \mathcal{T} \mid W \subseteq T\},$$

which we shall call $\mathcal{T}$ *pinched at* W. A family $\mathcal{T} \subseteq \mathfrak{T}$ is said to be ϵ*-concentrated at* w if $\Pr_{T \in \mathfrak{T}}[w \in T] \geq \epsilon$. We shall say that $\mathcal{T} \neq \emptyset$ is ϵ*-diffuse on* S, if for all $w \in S$, $\mathcal{T}$ is not ϵ-concentrated at w. A family $\mathcal{T}$ which is ϵ-concentrated at τ may be *pinched* at $\{\tau\}$, resulting in $\mathcal{T}_{\{\tau\}}$, with $|\mathcal{T}_{\{\tau\}}| \geq \epsilon|\mathcal{T}|$. An ϵ*-concentration sequence* for a family $\mathcal{T}$ is a sequence $\tau_1, \ldots, \tau_r$, such that

- $\mathcal{T}$ is ϵ-concentrated at τ_1,
- for $1 \leq i \leq r-1$, $\mathcal{T}_{\{\tau_1, \ldots, \tau_i\}}$ is ϵ-concentrated at τ_{i+1},
- $\mathcal{T}_{\{\tau_1, \ldots, \tau_r\}}$ is ϵ-diffuse on $\Sigma^{2n} - \{\tau_1, \ldots, \tau_r\}$.

LEMMA 3.2. *Let $\tau_1, \ldots, \tau_r$ be an ϵ-concentration sequence for $\mathcal{T} \subseteq \mathfrak{T}$. Then,*

$$r \leq \frac{-\log \mu(\mathcal{T})}{\log \epsilon + n},$$

where $\mu(\mathcal{T})$ is the density of $\mathcal{T}$ in $\mathfrak{T}$.

PROOF. Let $\tau = \{\tau_1, \ldots, \tau_r\}$. Since $\mathcal{T} \subseteq \mathfrak{T}$, $\mathcal{T}_\tau \subseteq \mathfrak{T}_\tau$ so that

$$\epsilon^r \mu(\mathcal{T}) \leq \mu(\mathcal{T}_\tau) \leq \mu(\mathfrak{T}_\tau) = 2^{-nr},$$

and hence

$$r \leq \frac{-\log \mu(\mathcal{T})}{\log \epsilon + n}.$$

□

THEOREM 3.3. *There exists an oracle O such that $\mathbf{mS}_2^{P,O} \subsetneq \mathbf{m}\Sigma_2^{P,O} \cap \mathbf{m}\Pi_2^{P,O}$.*

PROOF. For a pair of oracles $O_1, O_2 \subseteq \Sigma^*$, define

$$O_1 \oplus O_2 \stackrel{\text{def}}{=} \{0x \mid x \in O_1\} \cup \{1y \mid y \in O_2\}.$$

Let $\mathcal{C}$ be the set of all oracles $O = O_1 \oplus O_2$ such that for all $n \geq 0$,

$$\exists^n x\ \forall^n y\ xy \in O_1 \iff \forall^n x\ \exists^n y\ xy \in O_2.$$

Let $\mathcal{A}_n = \{C_0 \oplus C_1 \in \mathcal{C} \mid \exists^n x\ \forall^n y\ xy \in C_0\}$. Define

$$\mathcal{L}(O_1 \oplus O_2) = \{1^n \mid \exists^n x\ \forall^n y\ xy \in O_1\}$$

and notice that for $O \in \mathcal{C}$, we have that $\mathcal{L}(O) \in \mathbf{m}\Sigma_2^{P,O} \cap \mathbf{m}\Pi_2^{P,O}$. We shall construct an oracle $C = C_1 \oplus C_2 \in \mathcal{C}$ such that $\mathcal{L}(C) \notin \mathbf{m}S_2^{P,C}$. Let $\{D_i(\cdot,\cdot,\cdot)\}$ be an enumeration of monotone oracle S_2^P machines such that for every O, $\mathbf{m}S_2^{P,O} = \{L(D_i^O)\}$. Define $Q_i(n)$ to be the maximum size, over all oracles, x values, and y values, of any query made by $D_i(1^n, x, y)$. C shall be constructed in stages $C_1 \subseteq C_2 \subseteq \cdots$ such that $C = \bigcup_i C_i$, with stage i constructed to foil a specific monotone S_2^P machine. We shall have that

- for $i < j$, $C_i \subseteq C_j$ and $C_j - C_i \subseteq \Sigma^{k_j}$ for some k_j,
- $i < j \implies k_i < k_j$,
- for any oracle O with $O \cap \Sigma^{\leq k_s} = C_s$, $\mathcal{L}(O) \neq L(D_i^O)$ for any $i \leq s$.

Assume we have constructed the first $t-1$ stages, thus defining the oracle to length k_{t-1} and foiling the first $t-1$ machines. We shall construct C_t, foiling D_t. Let $p(n)$ be the running time of D_t. Select n such that $2n > k_{t-1}$, $2n > Q_{t-1}(k_{t-1})$, and $2^n > 2p(n)$. Set $k_t = 2n$. Assume, for contradiction, that regardless of our choice of $C_t \in \mathcal{C}$ (with $C_t \cap \Sigma^{\leq k_{t-1}} = C_{t-1}$), $D_t^{C_t}(1^n,\cdot,\cdot)$ accepts exactly when $C_t \in \mathcal{A}_n$.

For each $u \in \Sigma^n$, let $S_u \stackrel{\text{def}}{=} \{uv \mid |v| = n\}$, and consider the family of oracles

$$\{C_{t-1} \cup S_u \oplus T \mid T \in \mathfrak{T}\}.$$

Associate with each oracle O in this family an appropriate x, so that for all y, $D_t^O(1^n, x, y)$ accepts. There are at most $2^{p(n)}$ various values for x, so some x_u is associated with a fraction of this family of density at least $2^{-p(n)}$. Let $\mathcal{F}(u)$ be this (sub)family and define $\mathcal{T}(u) = \{T \mid C_{t-1} \cup S_u \oplus T \in \mathcal{F}_u\}$.

Let $\tau(u)_1, \tau(u)_2, \ldots, \tau(u)_{t(u)}$ be a $p(n)^{-1}$-concentration sequence for $\mathcal{T}(u)$ and $\tau(u) = \{\tau(u)_1, \tau(u)_2, \ldots, \tau(u)_{t(u)}\}$. By Lemma 3.2, the length of this sequence, $t(u)$, is at most

$$\frac{-\log \mu(\mathcal{T}(u))}{\log p(n)^{-1} + n} \leq \frac{p(n)}{n - \log p(n)} \leq p(n),$$

where the second inequality follows because $2^n > 2p(n)$. Reiterating, for each $u \in \Sigma^n$, we have selected a family of minterms $\mathcal{F}(u)$ all associated with a certain x_u and a $p(n)^{-1}$-concentration sequence $\tau(u)$ for $\mathcal{T}(u)$.

We similarly construct such sets of maxterms. For a subset $X \subseteq \Sigma^{2n}$, let $\hat{X} \stackrel{\text{def}}{=} \Sigma^{2n} - X$ be the relative complement of X. For each $v \in \Sigma^n$, consider

$$\{C_{t-1} \cup \hat{T} \oplus \hat{S}_v \mid T \in \mathfrak{T}\}.$$

As above, let y_v be associated with a family of the maxterms in the set $\mathcal{G}(v) \stackrel{\text{def}}{=} \{C_{t-1} \cup \hat{T} \oplus \hat{S}_v \mid T \in \mathcal{T}(v)\}$ such that $\mu(\mathcal{T}(v)) \geq 2^{-p(n)}$ (so that for any oracle O in this set and any x, $D_t^O(1^n, x, y_v)$ rejects). Let $\tau(v)_1, \ldots, \tau(v)_{t(v)}$ be a $p(n)^{-1}$-concentration sequence for $\mathcal{T}(v)$ and $\tau(v) = \{\tau(v)_1, \ldots, \tau(v)_{t(v)}\}$. Again $t(v) \leq p(n)$.

Selecting u and v uniformly and independently at random from Σ^n, we have that

$$\Pr_{u,v}[\exists i\ \tau(u)_i \in S_v \text{ or } \exists j\ \tau(v)_j \in S_u] \leq \frac{t(u) + t(v)}{2^n} \leq \frac{2p(n)}{2^n} < 1,$$

so that there exists a pair $(\tilde{u}, \tilde{v})$ with the property that

$$\forall i\ \tau(\tilde{u})_i \notin S_v, \qquad \forall j\ \tau(\tilde{v})_j \notin S_u.$$

Notice that an oracle O selected from $\mathcal{F}(\tilde{u}) \cup \mathcal{G}(\tilde{v})$ is accepted by $D_t^O(1^n, x_{\tilde{u}}, y_{\tilde{v}})$ exactly when $O \in \mathcal{F}(\tilde{u})$. Let $M = \Sigma^{2n} - \tau(\tilde{v}) \oplus \tau(\tilde{u})$ and consider the behavior of $D_t^M(1^n, x_{\tilde{u}}, y_{\tilde{v}})$. Suppose that $D_t^M(1^n, x_{\tilde{u}}, y_{\tilde{v}})$ accepts. Since D is monotone, every oracle $O \in \{C_{t-1} \cup \hat{T} \oplus \hat{S}_{\tilde{v}} \mid T \in \mathcal{T}(\tilde{v})_{\tau(\tilde{v})}\}$ must disagree with M in one of the places which $D_t^M(1^n, x_{\tilde{u}}, y_{\tilde{v}})$ queries at most $p(n)$ of them, which contradicts that $\mathcal{T}(\tilde{v})_{\tau(\tilde{v})}$ is $p(n)^{-1}$-diffuse on $\Sigma^{2n} - \tau(\tilde{v})$. The case when $D_t^M(1^n, x_{\tilde{u}}, y_{\tilde{v}})$ rejects is handled dually. □

4. Conclusions and open problems

In this note, we studied the notion of *symmetric alternation* by defining the complexity class $\mathbf{S}_2^P$. We observed certain structural properties of the $\mathbf{S}_2$ operator from which we glean some containment results. We show that $\mathbf{BPP} \subseteq \mathbf{S}_2^P$, by adapting the proof of Lautemann (1983) that $\mathbf{BPP} \subseteq \Sigma_2^P \cap \Pi_2^P$.

The original motivation for defining and studying the notion of symmetric alternation was a question posed by Uriel Feige. Independently, Canetti (1996) has studied an alternative notion of symmetric alternation and defined a class ϕ_2^P. He has shown that $\mathbf{BPP} \subseteq \phi_2^P$. In fact, an easy argument shows that $\mathbf{S}_2^P = \phi_2^P$. Feige *et al.* (1988) and J. Feigenbaum, D. Koller & Shor (1995) study situations, in an interactive setting, where the provers do not have complete access to each other's strategies. As a step towards characterizing the class of languages accepted by such interactive proof systems, we decided to formalize and study the associated non-interactive version in this paper.

It is an interesting open problem to construct an oracle separating $\mathbf{S}_2^P$ and $\Sigma_2^P \cap \Pi_2^P$ that would provide more evidence that the two classes are different. It would also be interesting to refine the coarse relationship expressed in Corollary 2.5 between the polynomial hierarchy and the $\mathbf{S}_2$ hierarchy.

Acknowledgements

We thank Mike Sipser, Oded Goldreich, and Alexander Shen for helpful discussions. The first author received support from an NSF Graduate Fellowship and grants NSF 92-12184, AFOSR F49620-92-J-0125, and DARPA N00014-92-1799. The second author received support from grants NSF 92-12184, AFOSR F49620-92-J-0125, and DARPA N00014-92-J-1799.

References

Miklós Ajtai, János Komlós, and E. Szemerédi, Deterministic simulation in LOGSPACE. In *Proceedings of the Nineteenth Annual ACM Symposium on the Theory of Computing*, 1987, 132–140.

Theodore P. Baker and Alan L. Selman, A second step toward the polynomial hierarchy. *Theoretical Computer Science* **8** (1979), 177–187.

R. Canetti, More on BPP and the polynomial-time hierarchy. *Information Processing Letters* **57**(5) (1996).

U. Feige, A. Shamir, and M. Tennenholtz, The noisy oracle problem. In *Proc. CPRYPTO 1988*, 1988, 284–296.

J. Feigenbaum, D. Koller and P. Shor, A game-theoretic characterization of interactive complexity classes (extended abstract). In *Proc. Structures '95*, Minneapolis, 1995, IEEE, 227–237.

John E. Hopcroft and Jeffrey D. Ullman, *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Series in Computer Science. Addison-Wesley, Reading, Massachusetts, 1979.

Russell Impagliazzo, Leonid A. Levin, and Michael Luby, Pseudo-random generation from one-way functions (extended abstract). In *Proceedings of the Twenty-first Annual ACM Symposium on the Theory of Computing*, 1989, 12–24.

C. Lautemann, BPP and the polynomial hierarchy. *Information Processing Letters* **17** (1983), 215–217.

Noam Nisan and David Zuckerman, More deterministic simulation in logspace. In *Proceedings of the Twenty-fifth Annual ACM Symposium on the Theory of Computing*, 1993, 235–244.

Michael Sipser, A complexity theoretic approach to randomness. In *Proceedings of the Fifteenth Annual ACM Symposium on the Theory of Computing*, 1983, 330–335.

L. J. Stockmeyer, The polynomial time hierarchy. *Theoretical Computer Science* **3** (1976), 1–22.

Andrew C. Yao, Theory and applications of trapdoor functions (extended abstract). In *Proceedings of the 23rd Annual Symposium on Foundations of Computer Science*, 1982, 80–91.

Manuscript received 16 March 1995

Alexander Russell
Department of Mathematics
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
`acr@theory.lcs.mit.edu`

Ravi Sundaram
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, MA 02139 USA
`koods@theory.lcs.mit.edu`

Current address of Alexander Russell:
Numerical Analysis and Computing Science
Royal Institute of Technology
S-100 44 Stockholm SWEDEN
`acr@nada.kth.se`

Current address of Ravi Sundaram:
Delta Global Trading L.P.
Four Cambridge Center
Cambridge, MA 02142 USA
`koods@delta-global.com`