Journal of Universal Computer Science, vol. 3, no. 3 (1997), 197-229
submitted: 4/6/96, accepted: 30/12/96, appeared: 28/3/97 © Springer Pub. Co.

Polynomial-Time Multi-Selectivity

Lane A. Hemaspaandra
(Dept. of Computer Science, University of Rochester, USA
lane@cs.rochester.edu)

Zhigen Jiang
(Institute of Software, Chinese Academy of Sciences, China
citicds!jason@uunet.uu.net)

Jörg Rothe
(Institut für Informatik, Friedrich-Schiller-Universität Jena, Germany
rothe@informatik.uni-jena.de)

Osamu Watanabe
(Dept. of Computer Science, Tokyo Institute of Technology, Japan
watanabe@cs.titech.ac.jp)

Abstract: We introduce a generalization of Selman's P-selectivity that yields a more flexible notion of selectivity, called (polynomial-time) multi-selectivity, in which the selector is allowed to operate on multiple input strings. Since our introduction of this class, it has been used [HJRW96] to prove the first known (and optimal) lower bounds for generalized selectivity-like classes in terms of EL_2, the second level of the extended low hierarchy. We study the resulting selectivity hierarchy, denoted by SH, which we prove does not collapse. In particular, we study the internal structure and the properties of SH and completely establish, in terms of incomparability and strict inclusion, the relations between our generalized selectivity classes and Ogihara's P-mc (polynomial-time membership-comparable) classes. Although SH is a strictly increasing infinite hierarchy, we show that the core results that hold for the P-selective sets and that prove them structurally simple also hold for SH. In particular, all sets in SH have small circuits; the NP sets in SH are in Low_2, the second level of the low hierarchy within NP; and SAT cannot be in SH unless $\mathrm{P} = \mathrm{NP}$. Finally, it is known that P-Sel, the class of P-selective sets, is not closed under union or intersection. We provide an extended selectivity hierarchy that is based on SH and that is large enough to capture those closures of the P-selective sets, and yet, in contrast with the P-mc classes, is refined enough to distinguish them.

Category: F.1.3

1 Introduction

Selman introduced the P-selective sets (P-Sel, for short) [Sel79] as the complexity-theoretic analogs of Jockusch's semi-recursive sets [Joc68]: A set is P-selective if there exists a polynomial-time transducer (henceforward called a selector) that, given any two input strings, outputs one that is logically no less likely to be in the set than the other one. There has been much progress recently in the study of P-selective sets (see the survey [DHHT94]). In this paper, we introduce a more flexible notion of selectivity that allows the selector to operate on multiple input strings, and that thus generalizes Selman's P-selectivity in the following promise-like way: Depending on two parameters, say i and j with $i \geq j \geq 1$, a set L is (i, j)-selective if there is a selector that, given any finite set of distinct input strings, outputs some subset of at least j elements each belonging to L if L contains at least i of the input strings; otherwise, it may output an arbitrary subset of the inputs. Observe that in this definition of (i,j)-selectivity only the difference of i and j is relevant: L is (i,j)-selective if and only if L is $(i-j+1,1)$-selective. Let $\mathrm{S}(k)$ denote the class of $(k,1)$-selective sets. Clearly, $\mathrm{S}(1) = \text{P-Sel}$, and for each $k \geq 1$, $\mathrm{S}(k) \subseteq \mathrm{S}(k+1)$. This paper is devoted to the study of the resulting hierarchy, $\mathrm{SH} \stackrel{\mathrm{df}}{=} \bigcup_{k \geq 1} \mathrm{S}(k)$.

The literature contains many notions that generalize P-selectivity. For example, Ko's "weakly P-selective sets" [Ko83], Amir, Beigel, and Gasarch's "non-p-superterse sets" [ABG90] (sometimes called "approximable sets" [BKS95]), Ogihara's "polynomial-time membership-comparable sets" [Ogi95], Cai and Hemaspaandra's (then Hemachandra) "polynomial-time enumerable sets" ([CH89], see the discussion in [Ogi95]), and the "$\mathcal{FC}$-selective sets for arbitrary function classes $\mathcal{FC}$" of Hemaspaandra et al. [HHN$^+$95] all are notions generalizing P-selectivity.

Given the number of already known and well-studied generalizations of P-Sel, the first question that naturally arises is: Why should one introduce another generalization of P-Sel? One motivation comes from other results of this paper's authors ([HJRW96], see also [Rot95]), which—in terms of the selectivity notion proposed in this paper—establish the first known (and optimal) lower bounds for generalized selectivity-like classes with regard to EL_2, the second level of the extended low hierarchy [BBS86]. In particular, there exists a sparse set in $\mathrm{S}(2)$ that is not in EL_2 [HJRW96, Rot95]. This sharply contrasts with the known result that all P-selective sets are in EL_2. The proof of this EL_2 lower bound additionally creates another interesting result: EL_2 is not closed under certain Boolean connectives such as union and intersection. This extends the known result that P-Sel is not closed under those Boolean connectives [HJ95]. Finally, the proof technique used to show the EL_2 lower bounds for generalized selectivity classes can be adapted to give the main result of [HJRW96]: There exist sets that are not in EL_2, yet their join is in EL_2. That is, the join operator can lower difficulty as measured in terms of extended lowness. Since in a strong intuitive sense the join does not lower complexity, this result suggests that, if one's intuition about complexity is—as is natural—based on reductions, then the extended low hierarchy is not a natural measure of complexity. Rather, it is a measure that is related to the difficulty of information extraction, and it is in flavor quite orthogonal to more traditional notions of complexity.

Another motivation for the study of the multi-selective sets is closely related to the known results mentioned in the previous paragraph. Since P-Sel is not closed under union or intersection, it is natural to ask which complexity classes are appropriate to capture, e.g., the class of intersections of P-selective sets. Even more to the point, can the intersections (or the unions) of P-selective sets be *classified* in some complexity-theoretic setting, for instance by proving that the class of intersections of P-selective sets is contained in such-and-such level of some hierarchy of complexity classes, but not in the immediately lower level? Though we will show that SH is not appropriate to provide answers to questions like this (since we prove that the above-mentioned result on unions and intersections extends to all levels of SH, i.e., neither the closure of P-Sel under union nor the closure of P-Sel under intersection is contained in any level of SH), we will introduce in Section 4 an extended selectivity hierarchy that is based on SH and can be used to classify Boolean closures of P-selective sets.

This paper is organized as follows. In Section 2, we provide our notations and some definitions. In Section 3.1, we study the internal structure and the properties of SH. In particular, we show that SH is properly infinite, and we relatedly prove that, unlike P-Sel, none of the S(k) for $k \geq 2$ is closed under $\leq_m^p$-reductions, and also that sets in S(2) that are many-one reducible to their complements may already go beyond P, which contrasts with Selman's result that a set A is in P if and only if $A \leq_m^p \overline{A}$ and A is P-selective [Sel79]. Consequently, the class P cannot be characterized by the auto-reducible sets in any of the higher levels of SH. This should be compared with Buhrman and Torenvliet's nice characterization of P as those self-reducible sets that are in P-Sel [BT96].

We then compare the levels of SH with the levels of Ogihara's hierarchy of polynomial-time membership-comparable (P-mc, for short) sets. Since P-mc(k) (see Definition 13) is closed under $\leq_{1\text{-}tt}^p$-reductions for each k [Ogi95], it is clear from the provable non-closure under $\leq_m^p$-reductions of the S(k), $k \geq 2$, that Ogihara's approach to generalized selectivity is different from ours, and in Theorem 14, we completely establish, in terms of incomparability and strict inclusion, the relations between his and our generalized selectivity classes. In particular, since P-mc(poly) is contained in P/poly [Ogi95] and SH is (strictly) contained in P-mc(poly), it follows that every set in SH has polynomial-size circuits. On the other hand, P-selective NP sets can even be shown to be in Low_2 [KS85]. Since such a result is not known to hold for the polynomial-time membership-comparable NP sets, our Low_2-ness results in Theorem 18 are the strongest known for generalized selectivity-like classes. (Note, however, that Köbler [Köb95] has observed that our generalization of Ko and Schöning's result that $\text{P-Sel} \cap \text{NP} \subseteq \mathrm{Low}_2$ [KS85] can be combined with other generalizations of the same result to yield a very generalized statement, as will be explained in more detail near the start of Section 3.2.)

Selman proved that NP-complete sets such as SAT (the satisfiability problem) cannot be P-selective unless P = NP [Sel79]. Ogihara extended this collapse result to the case of certain P-mc classes strictly larger than P-Sel. By the inclusions stated in Theorem 14, this extension applies to many of our selectivity classes as well; in particular, SH cannot contain all of NP unless P = NP.

To summarize, the results claimed in the previous two paragraphs (and to be proven in Section 3.2) demonstrate that the core results holding for the P-selective sets and proving them structurally simple also hold for SH.

In Section 4.1, we show into which levels of Ogihara's P-mc hierarchy the closures of P-Sel under certain Boolean operations fall. In particular, we prove that the closure of P-Sel under union and the closure of P-Sel under intersection fall into exactly the same level of the P-mc hierarchy and are not contained in the immediately lower level, which shows they are indistinguishable in terms of P-mc classes. We also show that the closure of P-Sel under certain Boolean operations is not contained in any level of SH. We then provide an extended selectivity hierarchy that is based on SH and is large enough to capture those closures of P-selective sets, and yet, in contrast with the P-mc classes, is refined enough to distinguish them. Finally, we study the internal structure of this extended selectivity hierarchy in Section 4.2. The proofs of some of the more technical results in Section 4.2 are deferred to Section 4.3.

2 Notations and Definitions

In general, we adopt the standard notations of Hopcroft and Ullman [HU79]. We consider sets of strings over the alphabet $\Sigma \stackrel{\text{df}}{=} \{0,1\}$. For each string $x \in \Sigma^*$, $|x|$ denotes the length of x. For $k \geq 1$, let $x^k \stackrel{\text{df}}{=} x \cdot x^{k-1}$, where $x^0 \stackrel{\text{df}}{=} \epsilon$ is the empty string and the dot denotes the concatenation of strings. $\mathcal{P}(\Sigma^*)$ is the class of sets of strings over Σ. Let $\mathbb{N}$ (respectively, $\mathbb{N}^+$) denote the set of non-negative (respectively, positive) integers. For any set $L \subseteq \Sigma^*$, $\|L\|$ represents the cardinality of L, and $\overline{L} \stackrel{\text{df}}{=} \Sigma^* - L$ denotes the complement of L in Σ^*.

For sets A and B, their *join*, $A \oplus B$, is $\{0x \mid x \in A\} \cup \{1x \mid x \in B\}$, and the Boolean operations *symmetric difference* (also called exclusive-or) and *equivalence* (also called nxor) are defined as $A \Delta B \stackrel{\text{df}}{=} (A \cap \overline{B}) \cup (\overline{A} \cap B)$ and $A \overline{\Delta} B \stackrel{\text{df}}{=} (A \cap B) \cup (\overline{A} \cap \overline{B})$. For any class $\mathcal{C}$, define $\mathrm{co}\mathcal{C} \stackrel{\text{df}}{=} \{L \mid \overline{L} \in \mathcal{C}\}$. For classes $\mathcal{C}$ and $\mathcal{D}$ of sets, define

$$\mathcal{C} \wedge \mathcal{D} \stackrel{\text{df}}{=} \{A \cap B \mid A \in \mathcal{C} \wedge B \in \mathcal{D}\},\ \mathcal{C}\, \boldsymbol{\Delta}\, \mathcal{D} \stackrel{\text{df}}{=} \{A \,\Delta\, B \mid A \in \mathcal{C} \wedge B \in \mathcal{D}\},$$
$$\mathcal{C} \vee \mathcal{D} \stackrel{\text{df}}{=} \{A \cup B \mid A \in \mathcal{C} \wedge B \in \mathcal{D}\},\ \mathcal{C}\, \overline{\boldsymbol{\Delta}}\, \mathcal{D} \stackrel{\text{df}}{=} \{A \,\overline{\Delta}\, B \mid A \in \mathcal{C} \wedge B \in \mathcal{D}\},$$
$$\mathcal{C} \bigoplus \mathcal{D} \stackrel{\text{df}}{=} \{A \oplus B \mid A \in \mathcal{C} \wedge B \in \mathcal{D}\}.$$

For k sets $A_1, \ldots, A_k$, the join extends to

$$\oplus_k(A_1, \ldots, A_k) \stackrel{\text{df}}{=} \bigcup_{1 \leq i \leq k} \{\underline{i}x \mid x \in A_i\},$$

where $\underline{i}$ is the bit pattern of $\lceil \log k \rceil$ bits representing i in binary. We write $\bigoplus_k(\mathcal{C})$ to denote the class $\{\oplus_k(A_1, \ldots, A_k) \mid (\forall i : 1 \leq i \leq k)\,[A_i \in \mathcal{C}]\}$ of k-ary joins of sets in $\mathcal{C}$. Similarly, we use the shorthands $\wedge_k(\mathcal{C})$ and $\vee_k(\mathcal{C})$ to denote the k-ary intersections and unions of sets in $\mathcal{C}$.

$L^{=n}$ (respectively, $L^{\leq n}$) is the set of strings in L having length n (respectively, less than or equal to n). Let $\Sigma^n \stackrel{\text{df}}{=} (\Sigma^*)^{=n}$. For a set L, χ_L denotes the characteristic function of L. The census function of L is defined by $\mathit{census}_L(0^n) \stackrel{\text{df}}{=} \|L^{\leq n}\|$. L is said to be sparse if there is a polynomial d such that for any n, $\mathit{census}_L(0^n) \leq d(n)$. Let SPARSE denote the class of sparse sets. To encode a pair of strings, we use a polynomial-time computable pairing function, $\langle \cdot, \cdot \rangle : \Sigma^* \times \Sigma^* \to \Sigma^*$, that has polynomial-time computable inverses; this notion is extended to encode every m-tuple of strings, in the standard way. Using the standard correspondence between Σ^* and $\mathbb{N}$, we will view $\langle \cdot, \cdot \rangle$ also as a pairing function mapping $\mathbb{N} \times \mathbb{N}$ onto $\mathbb{N}$. A polynomial-time transducer is a deterministic polynomial-time Turing machine that computes functions from Σ^* into Σ^* rather than accepting sets of strings. FP denotes the class of functions computed by polynomial-time transducers. Each selector function considered is computed by a polynomial-time transducer that takes a set of strings as input and outputs some set of strings. As the order of the strings in these sets doesn't matter, we may assume that, without loss of generality, they are given in lexicographical order (i.e., $x_1 \leq_{\text{lex}} x_2 \leq_{\text{lex}} \cdots \leq_{\text{lex}} x_m$), and are coded into one string over Σ using the above pairing function. As a notational convenience, we'll identify these sets with their codings and simply write (unless a more complete notation is needed) $f(x_1, \ldots, x_m)$ to indicate that selector f runs on the inputs $x_1, \ldots, x_m$ coded as $\langle x_1, \ldots, x_m \rangle$.

We shall use the shorthands NPM (NPOM) to refer to "nondeterministic polynomial-time (oracle) Turing machine." For an (oracle) Turing machine M (and an oracle set A), $L(M)$ ($L(M^A)$) denotes the set of strings accepted by M (relative to A). For any polynomial-time reducibility $\leq_r^p$ and any class of sets $\mathcal{C}$, define $\Re_r^p(\mathcal{C}) \stackrel{\mathrm{df}}{=} \{L \mid (\exists C \in \mathcal{C})\,[L \leq_r^p C]\}$. As is standard, E will denote $\bigcup_{c \geq 0} \mathrm{DTIME}[2^{cn}]$.

Definition 1. [KL80] P/poly denotes the class of sets L for which there exist a set $A \in \mathrm{P}$ and a polynomially length-bounded function $h : \Sigma^* \to \Sigma^*$ such that for every x, it holds that $x \in L$ if and only if $\langle x, h(0^{|x|}) \rangle \in A$.

Definition 2. 1. [Sch83] For $k \geq 1$, define $\mathrm{Low}_k \stackrel{\mathrm{df}}{=} \{L \in \mathrm{NP} \mid \Sigma_k^{p,L} = \Sigma_k^p\}$, where the Σ_k^p are the Σ levels of the polynomial hierarchy [MS72, Sto77].

2. [BBS86, LS95] For $k \geq 2$, define $\mathrm{EL}_k \stackrel{\mathrm{df}}{=} \{L \mid \Sigma_k^{p,L} = \Sigma_{k-1}^{p,\mathrm{SAT} \oplus L}\}$. For $k \geq 3$, define $\mathrm{EL}\Theta_k \stackrel{\mathrm{df}}{=} \{L \mid \mathrm{P}^{(\Sigma_{k-1}^{p,L})[\log n]} \subseteq \mathrm{P}^{(\Sigma_{k-2}^{p,\mathrm{SAT}\oplus L})[\log n]}\}$. The $[\log n]$ indicates that at most $\mathcal{O}(\log n)$ queries are made to the oracle.

3 A Basic Hierarchy of Generalized Selectivity Classes

3.1 Structure, Properties, and Relationships with P-mc Classes

Definition 3. Let g_1 and g_2 be non-decreasing functions from $\mathbb{N}^+$ into $\mathbb{N}^+$ (henceforward called *threshold functions*) such that $g_1 \geq g_2$. $\mathrm{S}(g_1(\cdot), g_2(\cdot))$ is the class of sets L for which there exists an FP function f such that for each $n \geq 1$ and any distinct input strings $y_1, \ldots, y_n$,

1. $f(y_1, \ldots, y_n) \subseteq \{y_1, \ldots, y_n\}$, and
2. if $\|L \cap \{y_1, \ldots, y_n\}\| \geq g_1(n)$, then it holds that $f(y_1, \ldots, y_n) \subseteq L$ and $\|f(y_1, \ldots, y_n)\| \geq g_2(n)$.

We also consider classes Fair-$\mathrm{S}(g_1(\cdot), g_2(\cdot))$ in which the selector f is required to satisfy the above conditions only when applied to any n distinct input strings each having length at most n. We will refer to selectors having this property as selectors meeting the "fairness condition."

As a notational convention and as a shorthand for describing functions, for *non*-constant threshold functions, we will use "expressions in n" and we use i, j, or k if the threshold is constant. For example, rather than writing $\mathrm{S}(\lambda n.n-1\,,\,\lambda n.k)$, we will use the shorthand $\mathrm{S}(n-1, k)$, and rather than writing $\mathrm{S}(\lambda n.g_1(n)\,,\,\lambda n.g_2(n))$ we will write $\mathrm{S}(g_1(n), g_2(n))$.

Definition 3 immediately implies the following:

Proposition 4. *Let g_1, g_2, and c be threshold functions such that $g_1 \geq g_2$.*

1. (a) $\mathrm{S}(g_1(n), g_2(n)) \subseteq \mathrm{S}(g_1(n) + c(n), g_2(n))$, *and*
(b) $\mathrm{S}(g_1(n), g_2(n) + c(n)) \subseteq \mathrm{S}(g_1(n), g_2(n))$.
The above inclusions also hold for the corresponding Fair-S classes.

2. *If* $(\forall m)[g_1(m) \geq m]$, *then* $\mathrm{S}(g_1(n), g_2(n)) = \text{Fair-S}(g_1(n), g_2(n)) = \mathcal{P}(\Sigma^*)$.
3. *If* $(\forall m)[g_2(m) \leq g_1(m) < m]$, *then* $\mathrm{S}(g_1(n), g_2(n)) \subseteq \text{Fair-S}(g_1(n), g_2(n)) \subseteq$ Fair-S$(n-1, 1)$.

In particular, we are interested in classes $\mathrm{S}(i, j)$ parameterized by constants i and j. Theorem 5 reveals that, in fact, there is only *one* significant parameter, the difference of i and j. This suggests the simpler notation $\mathrm{S}(k) \stackrel{\mathrm{df}}{=} \mathrm{S}(k, 1)$ for all $k \geq 1$. Let SH denote the hierarchy $\bigcup_{k \geq 1} \mathrm{S}(k)$. For simplicity, we henceforward (i.e., after the proof of Theorem 5) assume that selectors for any set in SH select *exactly one* input string rather than a subset of the inputs (i.e., they are viewed as FP functions mapping into Σ^* rather than into $\mathcal{P}(\Sigma^*)$).

Theorem 5. $(\forall i \geq 1)\,(\forall k \geq 0)\,[\mathrm{S}(i, 1) = \mathrm{S}(i + k, 1 + k)]$.

Proof. For any fixed $i \geq 1$, the proof is done by induction on k. The induction base is trivial. Assume $\mathrm{S}(i, 1) = \mathrm{S}(i + k - 1, k)$ for $k > 0$. We show that $\mathrm{S}(i, 1) = \mathrm{S}(i + k, 1 + k)$. For the first inclusion, assume $L \in \mathrm{S}(i, 1)$, and let f be an $\mathrm{S}(i + k - 1, k)$-selector for L that exists by the inductive hypothesis. Given any distinct input strings $y_1, \ldots, y_m$, $m \geq 1$, an $\mathrm{S}(i + k, 1 + k)$-selector g for L is defined by

$$g(y_1, \ldots, y_m) \stackrel{\mathrm{df}}{=} \begin{cases} f(\{y_1, \ldots, y_m\} - \{z\}) \cup \{z\} & \text{if } f(y_1, \ldots, y_m) \neq \emptyset \\ Y & \text{otherwise,} \end{cases}$$

where $z \in f(y_1, \ldots, y_m)$ and Y is an arbitrary subset of $\{y_1, \ldots, y_m\}$. Clearly, $g \in \mathrm{FP}$, $g(y_1, \ldots, y_m) \subseteq \{y_1, \ldots, y_m\}$, and if $\|L \cap \{y_1, \ldots, y_m\}\| \geq i + k$, then g outputs at least $1 + k$ strings each belonging to L. Thus, $L \in \mathrm{S}(i + k, 1 + k)$ via g.

For the converse inclusion, let $L \in \mathrm{S}(i + k, 1 + k)$ via g. To define an $\mathrm{S}(i + k - 1, k)$-selector f for L, let $i + k$ strings $z_1, \ldots, z_{i+k} \in L$ (w.l.o.g., L is infinite) be hard-coded into the machine computing f. Given $y_1, \ldots, y_m$ as input strings, $m \geq 1$, define

$$f(y_1, \ldots, y_m) \stackrel{\mathrm{df}}{=} \begin{cases} g(y_1, \ldots, y_m) & \text{if } \{z_1, \ldots, z_{i+k}\} \subseteq \{y_1, \ldots, y_m\} \\ g(y_1, \ldots, y_m, z) - \{z\} & \text{otherwise,} \end{cases}$$

where $z \in \{z_1, \ldots, z_{i+k}\} - \{y_1, \ldots, y_m\}$. Clearly, $f \in \mathrm{FP}$ selects a subset of its inputs $\{y_1, \ldots, y_m\}$, and if $\|L \cap \{y_1, \ldots, y_m\}\| \geq i + k - 1$, then f outputs at least k elements of L. Thus, f witnesses that $L \in \mathrm{S}(i + k - 1, k)$, which equals $\mathrm{S}(i, 1)$ by the inductive hypothesis. □

Proposition 6. *1.* $\mathrm{S}(1) = $ P-Sel.
2. $(\forall k \geq 1)\,[\mathrm{S}(k) \subseteq \mathrm{S}(k + 1)]$.

Proof. By definition, we have immediately Part 2 and the inclusion from left to right in Part 1, as in particular, given any pair of strings, an $\mathrm{S}(1)$-selector f is required to select a string (recall our assumption that all $\mathrm{S}(k)$-selectors output *exactly* one input string) that is no less likely to be in the set than the other one. For the converse inclusion, fix any set of inputs $y_1, \ldots, y_m$, $m \geq 1$, and let f be a P-selector for L. Play a knock-out tournament among the strings

$y_1, \dots, y_m$, where x beats y if and only if $f(x,y) = x$. Let y_w be the winner. Clearly, $g(y_1, \dots, y_m) \stackrel{\text{df}}{=} y_w$ witnesses that $L \in \mathrm{S}(1)$. □

Next we prove that SH is properly infinite and is strictly contained in Fair-S$(n-1,1)$. Recall that, by convention, the "$n-1$" in Fair-S$(n-1,1)$ denotes the non-constant threshold function $g(n) = n-1$. Fix an enumeration $\{f_i\}_{i\geq 1}$ of FP functions, and define $e(0) \stackrel{\text{df}}{=} 2$ and $e(k) \stackrel{\text{df}}{=} 2^{e(k-1)}$ for $k \geq 1$. For each $i \geq 0$ and $s \leq 2^{e(i)}$, let $W_{i,s} \stackrel{\text{df}}{=} \{w_{i,1}, \dots, w_{i,s}\}$ be an enumeration of the lexicographically smallest s strings in $\Sigma^{e(i)}$ (this notation will be used also in Section 4).

Theorem 7. *1. For each $k \geq 1$, $\mathrm{S}(k) \subset \mathrm{S}(k+1)$.*
2. SH $\subset$ Fair-S$(n-1,1)$.

Proof. 1. For fixed $k \geq 1$, choose $k+1$ pairwise distinct strings $b_0, \dots, b_k$ of the same length. Define

$$A_k \stackrel{\text{df}}{=} \bigcup_{i\geq 1} \left(\left\{ b_0^{e(i)}, \dots, b_k^{e(i)} \right\} - \left\{ f_i(b_0^{e(i)}, \dots, b_k^{e(i)}) \right\} \right),$$

i.e., for each $i \geq 1$, A_k can lack at most one out of the $k+1$ strings $b_0^{e(i)}, \dots, b_k^{e(i)}$.

An $\mathrm{S}(k+1)$-selector g for A_k is given in Figure 1. W.l.o.g., assume each input in $Y = \{y_1, \dots, y_m\}$ to be of the form $b_j^{e(i)}$ for some $j \in \{0, \dots, k\}$ and $i \in \{i_1, \dots, i_s\}$, where $1 \leq i_1 < \dots < i_s$ and $s \leq m$. Clearly, $g(Y) \in Y$. Let $n = |\langle y_1, \dots, y_m \rangle|$. Since there are at most m while loops to be executed and the polynomial-time transducers f_{i_t}, $t < s$, run on inputs of length at most $c \cdot \log e(i_s)$ for some constant c, the runtime of g on *that* input is bounded above by some polylogarithmic function in n. Then, there is a polynomial in n bounding g's runtime on *any* input. Thus, $g \in$ FP. If some element y is output during the while loop, then $y \in A_k$. If g outputs an arbitrary input string after exiting the while loop, then no input of the form $b_j^{e(i_t)}$, $t < s$, is in A_k, and since A_k has at most $k+1$ strings at each length, we have $\|A_k \cap Y\| \leq k$ if $g(Y) \notin A_k$. Thus, $A_k \in \mathrm{S}(k+1)$ via g.

On the other hand, each potential $\mathrm{S}(k)$-selector f_i, given $b_0^{e(i)}, \dots, b_k^{e(i)}$ as input strings, outputs an element not in A_k though k of these strings are in A_k. Thus, $A_k \notin \mathrm{S}(k)$.

2. Fix any $k \geq 1$, and let $L \in \mathrm{S}(k)$ via selector f. For each of the finitely many tuples $y_1, \dots, y_\ell$ such that $\ell \leq k$ and $|y_i| \leq \ell$, $1 \leq i \leq \ell$, let $z_{y_1,\dots,y_\ell}$ be some fixed string in $L \cap \{y_1, \dots, y_\ell\}$ if this set is non-empty, and an arbitrary string from $\{y_1, \dots, y_\ell\}$ otherwise. Let these fixed strings be hard-coded into the machine computing the function g defined by

$$g(y_1, \dots, y_n) \stackrel{\text{df}}{=} \begin{cases} \{z_{y_1,\dots,y_n}\} & \text{if } n \leq k \\ \{f(y_1, \dots, y_n)\} & \text{otherwise.} \end{cases}$$

Thus, $L \in$ Fair-S$(n-1,1)$ via g, showing that SH $\subseteq$ Fair-S$(n-1,1)$.

The strictness of the inclusion is proven as in Part 1 of this proof. To define a set $A \notin$ SH we have here to diagonalize against all potential selectors f_j and all levels of SH simultaneously. That is, in stage $i = \langle j, k \rangle$ of the construction

Description of an $\mathbf{S}(k+1)$-selector g.
  input $Y = \{y_1, \ldots, y_m\}$
  begin $t := s - 1$;
    while $t \geq 1$ **do**
      $Z := \{y \in Y \mid (\exists j \in \{0, \ldots, k\})\,[y = b_j^{e(i_t)}]\} - \{f_{i_t}(b_0^{e(i_t)}, \ldots, b_k^{e(i_t)})\}$;
      if $Z \neq \emptyset$ **then output** some element of Z and **halt**
      else $t := t - 1$
    end while
    output an arbitrary input string and **halt**
  end
End of description of g.

Figure 1: An $\mathrm{S}(k+1)$-selector g for A_k.

of $A \stackrel{\mathrm{df}}{=} \bigcup_{i \geq 1} A_i$, we will diagonalize against f_j being an $\mathrm{S}(k)$-selector for A. Fix $i = \langle j, k \rangle$. Recall that $W_{i,k+1}$ is the set of the smallest $k+1$ length $e(i)$ strings. Note that $2^{e(i)} \geq k+1$ holds for each i, since we can w.l.o.g. assume that the pairing function satisfies $u > \max\{v, w\}$ for all u, v, and w with $u = \langle v, w \rangle$. Define $A_i \stackrel{\mathrm{df}}{=} W_{i,k+1} - \{f_j(W_{i,k+1})\}$. Assume $A \in \mathrm{SH}$, i.e., there exists some t such that $A \in \mathrm{S}(t)$ via some selector f_s. But this contradicts that for $r = \langle s, t \rangle$, by construction of A, we have $\|A \cap W_{r,t+1}\| \geq t$, yet $f_s(W_{r,t+1})$ either doesn't output one of its inputs (and is thus no selector), or $f_s(W_{r,t+1}) \notin A$. Thus, $A \notin \mathrm{SH}$.

Now we prove that A trivially is in Fair-S$(n-1, 1)$, as A is constructed such that the promise is *never* met. By way of contradiction, suppose a set X of inputs is given, $\|X\| = n$, $\|A \cap X\| \geq n - 1$, and $|x| \leq n$ for each $x \in X$. Let $e(i)$ be the maximum length of the strings in $A \cap X$, i.e., $A \cap X = \bigcup_{m=1}^{i} A_m \cap X$. Let j and k be such that $i = \langle j, k \rangle$. Since (by the above remark about our pairing function) $k + 1 \leq i$, we have by construction of A,

$$e(i) - 1 \leq n - 1 \leq \|A \cap X\| = \|\bigcup_{m=1}^{i} A_m \cap X\| \leq \|\bigcup_{m=1}^{i} A_m\| \leq (k+1)i \leq i^2,$$

which is false for all $i \geq 0$. Hence, $A \in$ Fair-S$(n-1, 1)$. □

A variation of this technique proves that, unlike P-Sel, none of the $\mathrm{S}(k)$ for $k \geq 2$ is closed under $\leq_m^p$-reductions. (Of course, every class $\mathrm{S}(k)$ is closed downwards under polynomial-time one-one reductions.) We also show that sets in $\mathrm{S}(2)$ that are many-one reducible to their complements may already go beyond P, which contrasts with Selman's result that a set A is in P if and only if $A \leq_m^p \overline{A}$ and A is P-selective [Sel79]. It follows that the class P cannot be characterized by the auto-reducible sets (see [BT96]) in any of the higher classes in SH. It would be interesting to strengthen Corollary 9 to the case of the *self*-reducible sets, as that would contrast sharply with Buhrman and Torenvliet's characterization of P as those self-reducible sets that are in P-Sel [BT96].

Theorem 8. *1. For each $k \geq 2$, $\mathrm{S}(k) \subset \mathfrak{R}_m^p(\mathrm{S}(k))$.*

2. *There exists a set A in* $\mathrm{S}(2)$ *such that* $A \leq_m^p \overline{A}$ *and yet* $A \notin \mathrm{P}$.

Corollary 9. *There exists an auto-reducible set in* $\mathrm{S}(2)$ *that is not in* P.

Proof of Theorem 8. 1. In fact, for fixed k, we will define a set L in $\Re_m^p(\mathrm{S}(2)) - \mathrm{S}(k)$. By Fact 6, the theorem follows. Choose $2k$ pairwise distinct strings $b_1, \ldots, b_{2k}$ of the same length. Define $L \stackrel{\mathrm{df}}{=} \bigcup_{i\geq 1} A_i \cup B_i$, where

$$A_i \stackrel{\mathrm{df}}{=} \begin{cases} \{b_1^{e(i)}, \ldots, b_k^{e(i)}\} & \text{if } f_i(b_1^{e(i)}, \ldots, b_{2k}^{e(i)}) \notin \{b_1^{e(i)}, \ldots, b_k^{e(i)}\} \\ \emptyset & \text{otherwise,} \end{cases}$$

$$B_i \stackrel{\mathrm{df}}{=} \begin{cases} \{b_{k+1}^{e(i)}, \ldots, b_{2k}^{e(i)}\} & \text{if } f_i(b_1^{e(i)}, \ldots, b_{2k}^{e(i)}) \notin \{b_{k+1}^{e(i)}, \ldots, b_{2k}^{e(i)}\} \\ \emptyset & \text{otherwise.} \end{cases}$$

Clearly, each potential $\mathrm{S}(k)$-selector f_i, given $b_1^{e(i)}, \ldots, b_{2k}^{e(i)}$ as input strings, outputs an element not in L though $\|L \cap \{b_1^{e(i)}, \ldots, b_{2k}^{e(i)}\}\| \geq k$. Thus, $L \notin \mathrm{S}(k)$.

Now define the set

$$L' \stackrel{\mathrm{df}}{=} \{b_1^{e(i)} \mid b_1^{e(i)} \in L\} \cup \{b_{k+1}^{e(i)} \mid b_{k+1}^{e(i)} \in L\}$$

and an FP function g by $g(b_j^{e(i)}) \stackrel{\mathrm{df}}{=} b_1^{e(i)}$ if $1 \leq j \leq k$, and $g(b_j^{e(i)}) \stackrel{\mathrm{df}}{=} b_{k+1}^{e(i)}$ if $k+1 \leq j \leq 2k$, and $g(x) = x$ for all x not of the form $b_j^{e(i)}$ for any $i \geq 1$ and j, $1 \leq j \leq 2k$. Then, we have $x \in L$ if and only if $g(x) \in L'$ for each $x \in \Sigma^*$, that is, $L \leq_m^p L'$.

Now we show that $L' \in \mathrm{S}(2)$. Given any distinct inputs $y_1, \ldots, y_n$ (each having, without loss of generality, the form $b_1^{e(i)}$ or $b_{k+1}^{e(i)}$ for some $i \geq 1$), define an $\mathrm{S}(2)$-selector as follows:

Case 1: All inputs have the same length. Then, $\{y_1, \ldots, y_n\} \subseteq \{b_1^{e(i)}, b_{k+1}^{e(i)}\}$ for some $i \geq 1$. Define $f(y_1, \ldots, y_n)$ to be $b_1^{e(i)}$ if $b_1^{e(i)} \in \{y_1, \ldots, y_n\}$, and to be $b_{k+1}^{e(i)}$ otherwise. Hence, f selects a string in L' if $\|\{y_1, \ldots, y_n\} \cap L'\| \geq 2$.

Case 2: The input strings have different lengths. Let $\ell \stackrel{\mathrm{df}}{=} \max\{|y_1|, \ldots, |y_n|\}$. By brute force, we can decide in time polynomial in ℓ if there is some string with length smaller than ℓ in L'. If so, f selects the first string found. Otherwise, by the argument of Case 1, we can show that f selects a string (of maximum length) in L' if L' contains two of the inputs.

2. Let $\{M_i\}_{i\geq 1}$ be an enumeration of all deterministic polynomial-time Turing machines. Define

$$A \stackrel{\mathrm{df}}{=} \{0^{e(i)} \mid i \geq 1 \wedge 0^{e(i)} \notin L(M_i)\} \cup \{1^{e(i)} \mid i \geq 1 \wedge 0^{e(i)} \in L(M_i)\}.$$

Assume $A \in \mathrm{P}$ via M_j for some $j \geq 1$. This contradicts that $0^{e(j)} \in A$ if and only if $0^{e(j)} \notin L(M_j)$. Hence, $A \notin \mathrm{P}$. Define an FP function g by $g(0^{e(i)}) \stackrel{\mathrm{df}}{=} 1^{e(i)}$ and $g(1^{e(i)}) \stackrel{\mathrm{df}}{=} 0^{e(i)}$ for each $i \geq 1$; and for each $x \notin \{0^{e(i)}, 1^{e(i)}\}$, define $g(x) \stackrel{\mathrm{df}}{=} y$, where y is a fixed string in A (w.l.o.g., $A \neq \emptyset$). Clearly, $A \leq_m^p \overline{A}$ via g. $A \in \mathrm{S}(2)$ follows as above. □

Definition 10. For sets A and B, $A \leq_{m,\ell i}^{p} B$ if there is an FP function f such that for all $x \in \Sigma^*$, (a) $x \in A \iff f(x) \in B$, and (b) $x <_{\text{lex}} f(x)$.

Note that a similar kind of reduction was defined and was of use in [HHSY91], and that, intuitively, sets in $\{L \mid L \leq_{m,\ell i}^{p} L\}$ may be viewed as having a very weak type of padding functions.

Theorem 11. *If* $L \in \text{SH}$ *and* $L \leq_{m,\ell i}^{p} L$, *then* $L \in \text{P-Sel}$.

Proof. Let $L \leq_{m,\ell i}^{p} L$ via f, and let g be an $\text{S}(k)$-selector for L, for some k for which $L \in \text{S}(k)$. A P-selector h for L is defined as follows: Given any inputs x and y, generate two chains of k lexicographically increasing strings by running the reduction f, i.e., $x = x_1 <_{\text{lex}} x_2 <_{\text{lex}} \cdots <_{\text{lex}} x_k$ and $y = y_1 <_{\text{lex}} y_2 <_{\text{lex}} \cdots <_{\text{lex}} y_k$, where $x_2 = f(x)$, $x_3 = f(f(x))$, etc., and similarly for the y_i. To ensure that g will run on *distinct* inputs only (otherwise, g is not obliged to meet requirements 1 and 2 of Definition 3), let $z_1, \ldots, z_l$ be all the y_i's not in $\{x_1, \ldots, x_k\}$. Now run $g(x_1, \ldots, x_k, z_1, \ldots, z_l)$ and define $h(x, y)$ to output x if g outputs some string x_i, and to output y if g selects some string y_i (recall our assumption that $\text{S}(k)$-selectors such as g output exactly one string). Clearly, $h \in \text{FP}$, and if x or y are in L, then at least k inputs to g are in L, so h selects a string in L. □

Theorem 7 and Theorem 11 immediately imply the following:

Corollary 12. $\text{SH} \not\subseteq \{L \mid L \leq_{m,\ell i}^{p} L\}$.

Ogihara [Ogi95] has recently introduced the polynomial-time membership-comparable sets as another generalization of the P-selective sets.

Definition 13. [Ogi95] Let g be a monotone non-decreasing and polynomially bounded FP function from $\mathbb{N}$ to $\mathbb{N}^+$.

1. A function f is called a g-membership-comparing function (a g-mc-function, for short) for A if for every $z_1, \ldots, z_m$ with $m \geq g(\max\{|z_1|, \ldots, |z_m|\})$,
$$f(z_1, \ldots, z_m) \in \{0,1\}^m \text{ and } (\chi_A(z_1), \ldots, \chi_A(z_m)) \neq f(z_1, \ldots, z_m).$$
2. A set A is polynomial-time g-membership-comparable if there exists a polynomial-time computable g-mc-function for A.
3. $\text{P-mc}(g)$ denotes the class of polynomial-time g-membership-comparable sets.
4. $\text{P-mc(const)} \stackrel{\text{df}}{=} \bigcup\{\text{P-mc}(k) \mid k \geq 1\}$, $\text{P-mc(log)} \stackrel{\text{df}}{=} \bigcup\{\text{P-mc}(f) \mid f \in \mathcal{O}(\log)\}$, and $\text{P-mc(poly)} \stackrel{\text{df}}{=} \bigcup\{\text{P-mc}(p) \mid p \text{ is a polynomial}\}$.

Remark. We can equivalently (i.e., without changing the class) require in the definition that $f(z_1, \ldots, z_m) \neq (\chi_A(z_1), \ldots, \chi_A(z_m))$ must hold only if the inputs $z_1, \ldots, z_m$ happen to be *distinct*. This is true because if there are r and t with $r \neq t$ and $z_r = z_t$, then f simply outputs a length m string having a "0" at position r and a "1" at position t.

Since P-mc(k) is closed under $\leq^p_{1\text{-}tt}$-reductions for each k [Ogi95] but none of the S(k) for $k \geq 2$ is closed under $\leq^p_m$-reductions (Theorem 8), it is clear that Ogihara's approach to generalized selectivity is different from ours, and in Theorem 14 below, we completely establish, in terms of incomparability and strict inclusion, the relations between his and our generalized selectivity classes (see Figure 2). Note that Part 2 of Theorem 14 generalizes to k larger than 1 a result of Ogihara—who proved that the P-selective sets are strictly contained in P-mc(2) [Ogi95]—and the known fact that P-Sel is strictly larger than P [Sel79].

Theorem 14. *1.* P-mc(2) $\not\subseteq$ Fair-S($n-1, 1$).
2. For each $k \geq 1$, S(k) $\subset$ P-mc($k+1$) *and* S(k) $\not\subseteq$ P-mc(k).
3. S($n-1, 1$) $\subset$ P-mc(2).
4. Fair-S($n-1, 1$) $\subset$ P-mc(n) *and* Fair-S($n-1, 1$) $\not\subseteq$ P-mc($n-1$).

Proof. First recall that $\{f_i\}_{i\geq 1}$ is our enumeration of FP functions and that the set $W_{i,s} = \{w_{i,1}, \ldots, w_{i,s}\}$, for $i \geq 0$ and $s \leq 2^{e(i)}$, collects the lexicographically smallest s strings in $\Sigma^{e(i)}$, where function e is inductively defined to be $e(0) = 2$ and $e(i) = 2^{e(i-1)}$ for $i \geq 1$. Recall also our assumption that a selector for a set in SH outputs a single input string (if the promise is met), whereas S($n-1, 1$) and Fair-S($n-1, 1$) are defined via selectors that may output subsets of the given set of inputs.

1. We will construct a set A in stages. Let u_i be the smallest string in $W_{i,e(i)} \cap f_i(W_{i,e(i)})$ (if this set is non-empty; otherwise, f_i immediately disqualifies for being a Fair-S($n-1, 1$)-selector and we may go to the next stage). Define

$$A \stackrel{\text{df}}{=} \bigcup_{i\geq 1} (W_{i,e(i)} - \{u_i\}).$$

Then, $A \notin$ Fair-S($n-1, 1$), since for any i, $f_i(W_{i,e(i)})$ outputs a string not in A although $e(i) - 1$ of these inputs (each of length $e(i)$, i.e., the inputs satisfy the "fairness condition") *are* in A.

For defining a P-mc(2) function g for A, let any distinct inputs $y_1, \ldots, y_m$ with $m \geq 2$ be given. If there is some y_j such that $y_j \notin W_{i,e(i)}$ for each i, then define $g(y_1, \ldots, y_m)$ to be $0^{j-1}10^{m-j}$. If there is some y_j with $|y_j| < e(i_0)$, where $e(i_0) = \max\{|y_1|, \ldots, |y_m|\}$, then compute the bit $\chi_{\overline{A}}(y_j)$ by brute force in time polynomial in $e(i_0)$, and define $g(y_1, \ldots, y_m)$ to be $0^{j-1}\chi_{\overline{A}}(y_j)0^{m-j}$. Otherwise (i.e., if $\{y_1, \ldots, y_m\} \subseteq W_{i_0,e(i_0)}$), let $g(y_1, \ldots, y_m)$ be 0^m. Since, by definition of A, there is at most one string in $W_{i_0,e(i_0)}$ that is not in A, but $m \geq 2$, we have $g(y_1, \ldots, y_m) \neq (\chi_A(y_1), \ldots, \chi_A(y_m))$. Thus, $A \in$ P-mc(2) via g.

2. For fixed $k \geq 1$, let $L \in$ S(k) via f. Define a P-mc($k+1$) function g for L that, given distinct inputs $y_1, \ldots, y_m$ with $m \geq k+1$, outputs the string $1^{j-1}01^{m-j}$ if y_j is the string output by $f(y_1, \ldots, y_m)$. Clearly, $g(y_1, \ldots, y_m) \neq (\chi_L(y_1), \ldots, \chi_L(y_m))$, since there are at least k 1's in $1^{j-1}01^{m-j}$, and $f(y_1, \ldots, y_m) = y_j$ is thus a string in L. Hence, $L \in$ P-mc($k+1$) via g, showing S(k) $\subseteq$ P-mc($k+1$). By Statement 1, this inclusion is strict, and so is any inclusion to be proven below.

To show that S(k) $\not\subseteq$ P-mc(k), fix k strings $b_1, \ldots, b_k$ of the same length. Define

$$A \stackrel{\text{df}}{=} \left\{ b_j^{e(i)} \;\middle|\; \begin{array}{l} i \geq 1 \text{ and } f_i(b_1^{e(i)}, \ldots, b_k^{e(i)}) \in \{0,1\}^k \\ \text{and has a “1” at position } j,\ 1 \leq j \leq k \end{array} \right\}.$$

Clearly, since $f_i(b_1^{e(i)}, \ldots, b_k^{e(i)}) = (\chi_A(b_1^{e(i)}), \ldots, \chi_A(b_k^{e(i)}))$ for each i, no FP function f_i can serve as a P-mc(k) function for A. To define an S(k)-selector for A, let any inputs $y_1, \ldots, y_m$ (w.l.o.g., each of the form $b_j^{e(i)}$) be given, and let $\ell = \max\{|y_1|, \ldots, |y_m|\}$. As in the proofs of Theorem 7 and Theorem 8, it can be decided in time polynomial in ℓ whether there is some string of length smaller than ℓ in A. If so, the S(k)-selector f for A selects the first such string found. Otherwise, f outputs an arbitrary string of maximum length. Since there are at most k strings in A at any length, either the output string is in A, or $\|A \cap \{y_1, \ldots, y_m\}\| < k$. Thus, $\mathrm{S}(k) \not\subseteq \text{P-mc}(k)$. Statement 1 implies that as well $\text{P-mc}(k) \not\subseteq \mathrm{S}(k)$ for $k \geq 2$; the kth level of $\mathrm{SH} = \bigcup_{i \geq 1} \mathrm{S}(i)$ and the kth level of the hierarchy within P-mc(const) are thus incomparable.

3. Let $L \in \mathrm{S}(n-1, 1)$ via selector f. Define a P-mc(2) function g for L as follows: Given distinct input strings $y_1, \ldots, y_n$ with $n \geq 2$, g simulates $f(y_1, \ldots, y_n)$ and outputs the string $1^{j-1}01^{n-j}$ if y_j is any (say the smallest) string in $f(y_1, \ldots, y_n)$. Again, we can exclude one possibility for $(\chi_A(y_1), \ldots, \chi_A(y_n))$ via g in polynomial time, because the $\mathrm{S}(n-1, 1)$-promise is met for the string $1^{j-1}01^{n-j}$, and thus f must output a string in L.

4. Now we show that the proof of Statement 3 fails to some extent for the corresponding Fair-class, i.e., we will show that $\text{Fair-S}(n-1, 1) \not\subseteq \text{P-mc}(n-1)$. This resembles Part 2 of this theorem, but note that the proof now rests also on the "fairness condition" rather than merely on the $(n-1)$-promise. We also show that the "fairness condition" can no longer "protect" Fair-S$(n-1, 1)$ from being contained in P-mc(n).

$A \stackrel{\mathrm{df}}{=} \bigcup_{i \geq 1} A_i$ is defined in stages so that in stage i, f_i fails to be a P-mc$(n-1)$ function for A_i. This is ensured by defining A_i as a subset of the $e(i)-1$ smallest strings of length $e(i)$, $W_{i,e(i)-1}$, such that $w_{i,j} \in A_i$ if and only if $f_i(W_{i,e(i)-1})$ outputs a string of length $e(i)-1$ and has a "1" at position j. Thus, A is not in P-mc$(n-1)$, since $f_i(w_{i,1}, \ldots, w_{i,e(i)-1}) = (\chi_A(w_{i,1}), \ldots, \chi_A(w_{i,e(i)-1}))$ for each $i \geq 1$.

To see that $A \in \text{Fair-S}(n-1, 1)$, let any distinct inputs $y_1, \ldots, y_n$ be given, each having, w.l.o.g., length $e(i)$ for some i, and let $e(i_0)$ be their maximum length. As before, if there exists a string of length smaller than $e(i_0)$, say y_j, then it can be decided by brute force in polynomial time whether or not y_j belongs to A. Define a Fair-S$(n-1, 1)$-selector g to output $\{y_j\}$ if $y_j \in A$, and to output any input different from y_j if $y_j \notin A$. Thus, either the string output by g does belong to A, or $\|A \cap \{y_1, \ldots, y_n\}\| < n-1$. On the other hand, if all input strings are of the same length $e(i_0)$ and $\{y_1, \ldots, y_n\} \subseteq W_{i_0,e(i_0)-1}$, then the "fairness condition" is not fulfilled, as $e(i_0) > n$, and g is thus not obliged to output a string in A. If all inputs have length $e(i_0)$ and $\{y_1, \ldots, y_n\} \not\subseteq W_{i_0,e(i_0)-1}$, then by the above argument, g can be defined such that either the string output by g does belong to A, or $\|A \cap \{y_1, \ldots, y_n\}\| < n-1$. This completes the proof of $A \in \text{Fair-S}(n-1, 1)$.

Finally, we show that $\text{Fair-S}(n-1, 1) \subseteq \text{P-mc}(n)$. Let L be a set in Fair-S$(n-1, 1)$ via selector f. Let $y_1, \ldots, y_n$ be any distinct input strings such that $n \geq \max\{|y_1|, \ldots, |y_n|\}$, i.e., the "fairness condition" is now satisfied. Define a P-mc-function g for L which, on inputs $y_1, \ldots, y_n$, simulates $f(y_1, \ldots, y_n)$

and outputs the string $1^{j-1}01^{n-j}$ if f selects y_j. Thus,

$$g(y_1, \ldots, y_n) \neq (\chi_L(y_1), \ldots, \chi_L(y_n)),$$

and we have $L \in$ P-mc(n) via g. □

3.2 Circuit, Lowness, and Collapse Results

This section demonstrates that the core results (i.e., small circuit, Low_2-ness, and collapse results) that hold for the P-selective sets and that prove them structurally simple also hold for our generalized selectivity classes.

Since P-mc(poly) $\subseteq$ P/poly [Ogi95] and Fair-S$(n-1,1)$ is by Theorem 14 (strictly) contained in P-mc(n), it follows immediately that every set in Fair-S$(n-1,1)$ has polynomial-size circuits and is thus in $\mathrm{EL}\Theta_3$ (by Köbler's result that P/poly $\subseteq \mathrm{EL}\Theta_3$ [Köb94]). Note that Ogihara refers to Amir, Beigel, and Gasarch, whose P/poly proof for "non-p-superterse" sets (see [ABG90, Theorem 10]) applies to Ogihara's class P-mc(poly) as well. On the other hand, P-selective NP sets can even be shown to be in Low_2 [KS85], the second level of the low hierarchy within NP. In contrast, the proof of [ABG90, Theorem 10] does not give a Low_2-ness result for non-p-superterse NP sets, and thus also does not provide such a result for P-mc(poly) $\cap$ NP. By modifying the technique of Ko and Schöning, however, we generalize in Theorem 18 their result to our larger selectivity classes. Very recently, Köbler [Köb95] has observed that our generalization of Ko and Schöning's result that P-Sel $\cap$ NP $\subseteq \mathrm{Low}_2$ can be combined with others to yield a very generalized statement. In particular, he observed that our technique for proving Theorem 18 and the techniques used to prove results such as "any P-cheatable NP set is Low_2" [ABG90] and "any NPSV-selective NP set is Low_2" [HNOS96] are compatible. By combining the generalizing techniques simultaneously, Köbler can claim: Any NP set that is "strongly membership-comparable by NPSV functions" is Low_2 [Köb95]. (For the notations not defined here, we refer to [Köb95, ABG90, HNOS96].)

The proof of Theorem 18 explicitly constructs a family of non-uniform advice sets for any set in Fair-S$(n-1,1)$, as merely stating the existence of those advice sets (which follows from Theorem 15) does not suffice for proving Low_2-ness.

Note that some results of this section (e.g., Theorem 15) extend to the more general GC classes that will be defined in Section 4. We propose as an interesting task to explore whether all results of this section, in particular the Low_2-ness result of Theorem 18, apply to the GC classes.

Theorem 15. Fair-S$(n-1,1) \subseteq$ P/poly.

Corollary 16. SH $\subseteq$ P/poly.

Corollary 17. Fair-S$(n-1,1) \subseteq \mathrm{EL}\Theta_3$.

Theorem 18. *Any set in* NP $\cap$ Fair-S$(n-1,1)$ *is* Low_2.

Proof. Let L be any NP set in Fair-S$(n-1,1)$, and let f be a selector for L and N be an NPM such that $L = L(N)$. First, for each length m, we shall construct a polynomially length-bounded advice A_m that helps deciding membership of any string x, $|x| = m$, in L in polynomial time. For $m < 4$, take $A_m \stackrel{\mathrm{df}}{=} L^{=m}$ as advice. From now on let $m \geq 4$ be fixed, and let n be such that $4 \leq 2n \leq m$.

Some notations are in order. A subset G of $L^{=m}$ is called a *game* if $\|G\| = n$. Any output $w \in f(G)$ is called a *winner* of game G, and is said to be *yielded* by the *team* $G - \{w\}$. If $\|L^{=m}\| \leq 2(n+1)$, then simply take $A_m \stackrel{\mathrm{df}}{=} L^{=m}$ as advice. Otherwise, A_m is constructed in *rounds*. In round i, one team, t_i, is added to A_m, and all winners yielded by that team in any game are deleted from a set B_{i-1}. Initially, B_0 is set to be $L^{=m}$.

In more detail, in the first round, all games of $B_0 = L^{=m}$, one after the other, are fed into the selector f for L to determine all winners of each game, and, associated with each winner, the team yielding that winner. We will argue below that there must exist at least one team yielding at least $\frac{\binom{N}{n}}{\binom{N}{n-1}}$ winners if N is the number of strings in $L^{=m}$. Choose the "smallest" (according to the ordering $\leq_{\mathrm{lex}}$ on $L^{=m}$) such team, t_1, and add it to the advice A_m. Delete from B_0 all winners yielded by t_1 and set B_1 to be the remainder of B_0, i.e.,

$$B_1 \stackrel{\mathrm{df}}{=} B_0 - \{w \mid \text{winner } w \text{ is yielded by team } t_1\},$$

and, entering the second round, repeat this procedure with all games of B_1 unless B_1 has $\leq 2(n+1)$ elements. In the second round, a second team t_2, and in later rounds more teams t_i, are determined and are added to A_m. The construction of A_m in rounds will terminate if $\|B_{k(m)}\| \leq 2(n+1)$ for some integer $k(m)$ depending on the given length m. In that case, add $B_{k(m)}$ to A_m. Formally,

$$A_m \stackrel{\mathrm{df}}{=} B_{k(m)} \cup \bigcup_{i=1}^{k(m)} t_i,$$

where $B_{k(m)} \subseteq L^{=m}$ contains at most $2(n+1)$ elements, $t_i \subseteq L^{=m}$ is the team added to A_m in round i, $1 \leq i \leq k(m)$, and the bound $k(m)$ on the number of rounds executed at length m is specified below.

We now show that there is some polynomial in m bounding the length of (the coding of) A_m for any m. If $L^{=m}$ has $N > 2(n+1)$ strings, then there are $\binom{N}{n}$ games and $\binom{N}{n-1}$ teams in the first round. Since every game has at least one winner, there exists one team yielding at least

$$\frac{\binom{N}{n}}{\binom{N}{n-1}} = \frac{N-n+1}{n} > \frac{N}{2n} \geq \frac{N}{m}$$

winners to be deleted from B_0 in the first round. Thus, there remain in B_1 at most $N\left(1 - \frac{1}{m}\right)$ elements after the first round, and, successively applying this argument, B_k contains at most $N\left(1 - \frac{1}{m}\right)^k$ elements after k rounds. Since $N \leq 2^m$ and the procedure terminates if $\|B_k\| \leq 2(n+1)$ for some integer k, it suffices to show that some polynomial $k(m)$ of fixed degree satisfies

$$\left(1 - \frac{1}{m}\right)^{k(m)} \leq 2(n+1)2^{-m}.$$

This follows from the fact that $\lim_{m\to\infty}\left(\left(1-\frac{1}{m}\right)^{m^2}\right)^{m^{-1}} = e^{-1} < \frac{1}{2}$ implies that $\left(1-\frac{1}{m}\right)^{m^2} = \mathcal{O}(2^{-m})$. As in each round $n-1 < m$ strings of length m are added to A_m, the length of (the coding of) A_m is indeed bounded above by some polynomial of degree 4.

Note that the set

$$C \stackrel{\mathrm{df}}{=} \left\{ \langle x, a_{|x|}\rangle \;\middle|\; \begin{array}{l} a_{|x|} \text{ is encoding of an advice } A_{|x|} \text{ and } x \in B_{k(|x|)}, \text{ or } (\exists t_j) \\ {[t_j \text{ is a team of } A_{|x|} \text{ and } x \text{ belongs to or is yielded by } t_j]} \end{array} \right\}$$

witnesses $L \in \mathrm{P/poly}$ (as stated in Theorem 15), since clearly C is a set in P and $L = \{x \mid \langle x, a_{|x|}\rangle \in C\}$.

Now we are ready to prove $L \in \mathrm{Low}_2$. Let $D \in \mathrm{NP}^{\mathrm{NP}^L}$ be witnessed by some NPOMs N_1 and N_2, that is, $D = L(N_1^{L(N_2^L)})$. Let $q(\ell)$ be a polynomial bound on the length of all queries that can be asked in this computation on an input of length ℓ. We describe below an NPOM M and an NP oracle set E for which $D = L(M^E)$.

On input x, M guesses for each length m, $1 \leq m \leq q(|x|)$, all possible polynomially length-bounded advice sets A_m for $L^{=m}$, simultaneously guessing witnesses (that is, an accepting path of N on input z) that each string z in any guessed advice set is in $L^{=m}$. To check on each path whether the guessed sequence of advice sets is correct, M queries its oracle E whether it contains the string $\langle x, A_1, \ldots, A_{q(|x|)}\rangle$, where

$$E \stackrel{\mathrm{df}}{=} \left\{ \langle x, A_1, \ldots, A_{q(|x|)}\rangle \;\middle|\; \begin{array}{l} (\exists m : 1 \leq m \leq q(|x|))\,(\exists y_m : |y_m| = m)\,(\exists w_m)\,[w_m \\ \text{is an accepting path of } N(y_m), \text{ yet } y_m \text{ is neither} \\ \text{a string in } A_m \text{ nor is yielded by any team of } A_m] \end{array} \right\}$$

is clearly a set in NP. If the answer is "yes," then some guessed advice is incorrect, and M rejects on that computation. If the answer is "no," then each guessed advice is correct for any possible query of the respective length. Thus, M now can simulate the computation of $N_1^{L(N_2)}$ on input x using the selector f and the relevant advice A_m to answer any question of N_2 correctly. Hence, $D \in \mathrm{NP}^{\mathrm{NP}}$. □

Ogihara has shown that if $\mathrm{NP} \subseteq \text{P-mc}(c\log n)$ for some $c < 1$, then $\mathrm{P} = \mathrm{NP}$ [Ogi95]. Since by the proof of Theorem 14, Fair-S$(c\log n, 1)$ is contained in P-mc$(c\log n)$, $c < 1$, we have immediately the following corollary to Ogihara's result. (Although Ogihara's result in [Ogi95] is also established for certain complexity classes other than NP, we focus on the NP case only.)

Corollary 19. *If* $\mathrm{NP} \subseteq \text{Fair-S}(c\log n, 1)$ *for some* $c < 1$*, then* $\mathrm{P} = \mathrm{NP}$.

4 An Extended Selectivity Hierarchy Capturing Boolean Closures of P-Selective Sets

4.1 Distinguishing Between and Capturing Boolean Closures of P-Selective Sets

Hemaspaandra and Jiang [HJ95] noted that the class P-Sel is closed under exactly those Boolean connectives that are either completely degenerate or almost-completely degenerate. In particular, P-Sel is not closed under intersection or union, and is not even closed under marked union (join). This raises the question of how complex, e.g., the intersection of two P-selective sets is. Also, is the class of unions of two P-selective sets more or less complex than the class of intersections of two P-selective sets? Theorem 24 establishes that, in terms of P-mc classes, unions and intersections of sets in P-Sel are indistinguishable (though they both are different from exclusive-or). However, we will note as Theorem 25 that the GC hierarchy (defined below) does distinguish between these classes, thus capturing the closures of P-Sel under certain Boolean connectives more tightly.

Definition 20. Let g_1, g_2, and g_3 be threshold functions.

Define $\mathrm{GC}(g_1(\cdot), g_2(\cdot), g_3(\cdot))$ to be the class of sets L for which there exists a polynomial-time computable function f such that for each $n \geq 1$ and any distinct input strings $y_1, \ldots, y_n$,

1. $f(y_1, \ldots, y_n) \subseteq \{y_1, \ldots, y_n\}$ and $\|f(y_1, \ldots, y_n)\| \leq g_2(n)$, and
2. $\|L \cap \{y_1, \ldots, y_n\}\| \geq g_1(n) \implies \|L \cap f(y_1, \ldots, y_n)\| \geq g_3(n)$.

Remark. 1. The notational conventions described after Definition 3 also apply to Definition 20.

2. For constant thresholds b, c, d, we can equivalently (i.e., without changing the class) require in the definition that the selector f for a set L in $\mathrm{GC}(b, c, d)$, on all input sets of size at least c, must output *exactly* c strings. This is true because if f outputs fewer than c strings, we can define a new selector f' that outputs all strings output by f and additionally $\|f\| - c$ arbitrary input strings not output by f, and f' is still a $\mathrm{GC}(b, c, d)$-selector for L. This will be useful in the proof of Lemma 30.

The GC classes generalize the S classes of Section 3, and as before, we also consider Fair-GC classes by additionally requiring the "fairness condition." Let GCH denote $\bigcup_{i,j,k \geq 1} \mathrm{GC}(i, j, k)$. The internal structure of GCH will be analyzed in Section 4.2.

A class $\mathcal{C} \subseteq \mathcal{P}(\Sigma^*)$ of sets is said to be *nontrivial* if $\mathcal{C}$ contains infinite sets, but not all sets of strings over Σ. For example, the class Fair-GC$(\lceil \frac{n}{2} \rceil, \lceil \frac{n}{2} \rceil, 1)$ equals $\mathcal{P}(\Sigma^*)$ if n is odd, and is therefore called trivial. First we note below that the largest nontrivial GC class, Fair-GC$(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, 1)$, and thus all of GCH, is contained in the P-mc hierarchy.

Theorem 21. Fair-GC$(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, 1) \subseteq$ P-mc(*poly*).

Proof. Let $L \in$ Fair-GC$(\lfloor \frac{n}{2} \rfloor, \lfloor \frac{n}{2} \rfloor, 1)$ via selector f. Fix any distinct inputs $y_1, \ldots, y_n$ such that $n \geq (\max\{|y_1|, \ldots, |y_n|\})^2$. Define a P-mc$(n^2)$ function g as follows: g simulates $f(y_1, \ldots, y_n)$ and outputs a "0" at each position corresponding to an output string of f, and outputs a "1" anywhere else. If all the strings having a "1" in the output of g indeed are in L, then at least one of the outputs of f must be in L, since the "fairness condition" is met and $\|\{y_1, \ldots, y_n\} \cap L\| \geq \frac{n}{2}$. Thus,

$$(\chi_L(y_1), \ldots, \chi_L(y_n)) \neq g(y_1, \ldots, y_n),$$

and we have $L \in$ P-mc(poly) via g. ☐

Now we state two lemmas that will be useful in the upcoming proofs of Theorem 24 and Theorem 25.

Lemma 22. [BT96] *Let $A \in$ P-Sel and $V \subseteq \Sigma^*$. The P-selector f for A induces a total order $\preceq_f$ on V as follows: For each x and y in V, define $x \preceq_f y$ if and only if*

$$(\exists u_1, \ldots, u_k)\,[x = u_1 \wedge y = u_k \wedge (\forall i : 2 \leq i \leq k)\,[f(u_{i-1}, u_i) = u_i]].$$

Then, for all $x, y \in V$,

$$x \preceq_f y \iff (x \in A \implies y \in A).$$

The technique of constructing *widely-spaced* and *complexity-bounded* sets is a standard technique for constructing P-selective sets. This technique will be useful in the diagonalization proofs of this section and will be applied in the form presented in [HJ95, HJRW96, Rot95]. So let us first adopt some of the formalism used in these papers.

Fix some wide-spacing function μ such that the spacing is at least as wide as given by the following inductive definition: $\mu(0) = 2$ and $\mu(i+1) = 2^{2^{\mu(i)}}$ for each $i \geq 0$. Now define for each $k \geq 0$,

$$R_k \stackrel{\text{df}}{=} \{i \mid i \in \mathbb{N} \wedge \mu(k) \leq i < \mu(k+1)\},$$

and the following two classes of languages (where we will implicitly use the standard correspondence between Σ^* and $\mathbb{N}$):

$$\mathcal{C}_1 \stackrel{\text{df}}{=} \left\{ A \subseteq \mathbb{N} \;\middle|\; \begin{array}{l} (\forall j \geq 0)\,[R_{2j} \cap A = \emptyset \wedge (\forall x, y \in R_{2j+1}) \\ \;[(x \leq y \wedge x \in A) \implies y \in A]] \end{array} \right\};$$

$$\mathcal{C}_2 \stackrel{\text{df}}{=} \left\{ A \subseteq \mathbb{N} \;\middle|\; \begin{array}{l} (\forall j \geq 0)\,[R_{2j} \cap A = \emptyset \wedge (\forall x, y \in R_{2j+1}) \\ \;[(x \leq y \wedge y \in A) \implies x \in A]] \end{array} \right\}.$$

Then, the following lemma can be proven in the same vein as in [HJ95].

Lemma 23. [HJ95] $\mathcal{C}_1 \cap \mathrm{E} \subseteq$ P-Sel *and* $\mathcal{C}_2 \cap \mathrm{E} \subseteq$ P-Sel.

Remark. 1. We will apply Lemma 23 in a slightly more general form in the proof of Theorem 24 below. That is, in the definition of $\mathcal{C}_1$ and $\mathcal{C}_2$, the underlying ordering of the elements in the regions R_{2j+1} need not be the standard lexicographical order of strings. We may allow *any* ordering $\prec$ that respects the lengths of strings and such that, given two strings, x and y, of the same length, it can be decided in polynomial time whether $x \prec y$.

2. To accomplish the diagonalizations in this section, we need our enumeration of FP functions to satisfy a technical requirement. Fix an enumeration of all polynomial-time transducers $\{T_i\}_{i \geq 1}$ having the property that each transducer appears infinitely often in the list. That is, if $T = T_i$ (here, equality refers to the actual program) for some i, then there is an infinite set J of distinct integers such that for each $j \in J$, we have $T = T_j$. For each $k \geq 1$, let f_k denote the function computed by T_k. In the diagonalizations below, it is enough to diagonalize for all k against some $T_{k'}$ such that $T_k = T_{k'}$, i.e., both compute f_k. In particular, for keeping the sets L_1 and L_2 (to be defined in the upcoming proofs of Theorems 24 and 25) in E, we will construct L_1 and L_2 such that for all stages j of the construction and for any set of

inputs $X \subseteq R_{2j+1}$, the transducer computing $f_j(X)$ runs in time less than $2^{\max\{|x|\,:\,x\in X\}}$ (i.e., the simulation of T_j on input X is aborted if it fails to be completed in this time bound, and the construction of L_1 and L_2 proceeds to the next stage). The diagonalization is still correct, since for each T_i there is a number b_i (depending only on T_i) such that for each $k \geq b_i$, if $T_i = T_k$, then for T_k we will properly diagonalize—and thus T_i is implicitly diagonalized against.

3. For each $j \geq 0$ and $k < \|R_{2j+1}\|$, let $r_{j,0}, \ldots, r_{j,k}$ denote the strings corresponding to the first $k+1$ numbers in region R_{2j+1} (in the standard correspondence between Σ^* and $\mathbb{N}$).

Theorem 24. *1.* P-Sel $\wedge$ P-Sel $\subseteq$ P-mc(3), *yet* P-Sel $\wedge$ P-Sel $\not\subseteq$ P-mc(2).
2. P-Sel $\vee$ P-Sel $\subseteq$ P-mc(3), *yet* P-Sel $\vee$ P-Sel $\not\subseteq$ P-mc(2).
3. P-Sel $\boldsymbol{\Delta}$ P-Sel $\not\subseteq$ P-mc(3) *and* P-Sel $\overline{\boldsymbol{\Delta}}$ P-Sel $\not\subseteq$ P-mc(3).

Proof. 1. & 2. Let $A \in$ P-Sel via f and $B \in$ P-Sel via g, and let $\preceq_f$ and $\preceq_g$ be the orders induced by f and g, respectively. Fix any inputs y_1, y_2, and y_3 such that $y_1 \preceq_f y_2 \preceq_f y_3$. Define a P-mc(3) function h for $A \cap B$ as follows. If f and g "agree" on any two of these strings (i.e., if there exist $i, j \in \{1,2,3\}$ such that $i < j$ and $y_i \preceq_g y_j$), then $h(y_1, y_2, y_3)$ outputs a "1" at position i and a "0" at position j. Otherwise (i.e., if $y_3 \preceq_g y_2 \preceq_g y_1$), define $h(y_1, y_2, y_3)$ to output the string 101. In each case, we have

$$(\chi_{A\cap B}(y_1), \chi_{A\cap B}(y_2), \chi_{A\cap B}(y_3)) \neq h(y_1, y_2, y_3).$$

A similar construction works for $A \cup B$: Define $h(y_1, y_2, y_3)$ to output the string 010 if $y_3 \preceq_g y_2 \preceq_g y_1$, and as above in the other cases. This proves P-Sel $\wedge$ P-Sel $\subseteq$ P-mc(3) and P-Sel $\vee$ P-Sel $\subseteq$ P-mc(3).

For proving the diagonalizations, recall from the remark after Lemma 23 that $r_{j,0}, \ldots, r_{j,k}$ denote the smallest $k+1$ numbers in region R_{2j+1}. Define $L_1 \stackrel{\text{df}}{=} \bigcup_{j\geq 0} L_{1,j}$ and $L_2 \stackrel{\text{df}}{=} \bigcup_{j\geq 0} L_{2,j}$, where

$$L_{1,j} \stackrel{\text{df}}{=} \left\{ i \in R_{2j+1} \;\middle|\; \begin{array}{l} (f_j(r_{j,0}, r_{j,1}) \in \{00, 01\} \wedge i \geq r_{j,1}) \vee \\ (f_j(r_{j,0}, r_{j,1}) \in \{10, 11\} \wedge i \geq r_{j,0}) \end{array} \right\};$$

$$L_{2,j} \stackrel{\text{df}}{=} \left\{ i \in R_{2j+1} \;\middle|\; \begin{array}{l} (f_j(r_{j,0}, r_{j,1}) \in \{00, 10\} \wedge i \leq r_{j,0}) \vee \\ (f_j(r_{j,0}, r_{j,1}) \in \{01, 11\} \wedge i \leq r_{j,1}) \end{array} \right\}.$$

Clearly, by the above remark about the construction of L_1 and L_2, we have that L_1 is in $\mathcal{C}_1 \cap \mathrm{E}$ and L_2 is in $\mathcal{C}_2 \cap \mathrm{E}$. Thus, by Lemma 23, L_1 and L_2 are in P-Sel. Supposing $L_1 \cap L_2 \in$ P-mc(2) via f_{j_0} for some j_0, we have a string $f_{j_0}(r_{j_0,0}, r_{j_0,1})$ in $\{0,1\}^2$ that satisfies:

$$(\chi_{L_1\cap L_2}(r_{j_0,0}), \chi_{L_1\cap L_2}(r_{j_0,1})) \neq f_{j_0}(r_{j_0,0}, r_{j_0,1}).$$

However, in each of the four cases for the membership of $r_{j_0,0}$ and $r_{j_0,1}$ in $L_1 \cap L_2$, this is by definition of L_1 and L_2 exactly what f_{j_0} claims is impossible. Therefore, P-Sel $\wedge$ P-Sel $\not\subseteq$ P-mc(2). Furthermore, since P-Sel is closed under complementation, $\overline{L_1}$ and $\overline{L_2}$ are in P-Sel. Now assume P-Sel $\vee$ P-Sel $\subseteq$ P-mc(2).

Then, $\overline{L_1} \cup \overline{L_2} = \overline{L_1 \cap L_2}$ is in P-mc(2), and since P-mc(2) is closed under complementation, we have $L_1 \cap L_2 \in$ P-mc(2), a contradiction. Hence, P-Sel $\vee$ P-Sel $\not\subseteq$ P-mc(2).

3. Let $L_1 \stackrel{\mathrm{df}}{=} \bigcup_{j\geq 0} L_{1,j}$, where $L_{1,j}$ is the set of all $i \in R_{2j+1}$ such that

(a) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) \in \{100, 101, 111\} \wedge i \geq r_{j,0})$ or
(b) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) = 011 \wedge i \geq r_{j,1})$ or
(c) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) \in \{001, 110\} \wedge i \geq r_{j,2})$.

Thus, $L_1 \in \mathcal{C}_1 \cap$ E, and by Lemma 23, $L_1 \in$ P-Sel.

For defining L_2, let us first assume the following reordering of the elements in R_{2j+1} for each $j \geq 0$: $r_{j,1} \prec r_{j,2} \prec r_{j,0} \prec r_{j,3}$ and $r_{j,s} \prec r_{j,s+1}$ if and only if $r_{j,s} < r_{j,s+1}$ for $s \geq 3$. For any strings x and y, we write $x \preceq y$ if $x \prec y$ or $x = y$. Now define $L_2 \stackrel{\mathrm{df}}{=} \bigcup_{j\geq 0} L_{2,j}$, where $L_{2,j}$ is the set of all $i \in R_{2j+1}$ such that

(a) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) = 110 \wedge i \preceq r_{j,0})$ or
(b) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) \in \{010, 101\} \wedge i \preceq r_{j,1})$ or
(c) $(f_j(r_{j,0}, r_{j,1}, r_{j,2}) = 100 \wedge i \preceq r_{j,2})$.

By Lemma 23 and the remark following Lemma 23, $L_2 \in$ P-Sel. Note that for each $j \geq 0$, the set $L_1 \cap R_{2j+1}$ is empty if $f_j(r_{j,0}, r_{j,1}, r_{j,2}) \in \{000, 010\}$, and the set $L_2 \cap R_{2j+1}$ is empty if $f_j(r_{j,0}, r_{j,1}, r_{j,2})$ is in $\{000, 001, 011, 111\}$. Now suppose $L_1 \Delta L_2 \in$ P-mc(3) via f_{j_0} for some j_0, i.e., $f_{j_0}(r_{j_0,0}, r_{j_0,1}, r_{j_0,2})$ is in $\{0,1\}^3$ and satisfies

$$(\chi_{L_1 \Delta L_2}(r_{j_0,0}), \chi_{L_1 \Delta L_2}(r_{j_0,1}), \chi_{L_1 \Delta L_2}(r_{j_0,2})) \neq f_{j_0}(r_{j_0,0}, r_{j_0,1}, r_{j_0,2}).$$

However, in each of the eight cases for the membership of $r_{j_0,0}$, $r_{j_0,1}$, and $r_{j_0,2}$ in $L_1 \Delta L_2$, this is by definition of L_1 and L_2 exactly what f_{j_0} claims is impossible. Therefore, P-Sel $\boldsymbol{\Delta}$ P-Sel $\not\subseteq$ P-mc(3). Since $L_1 \overline{\Delta} \overline{L_2} = L_1 \Delta L_2$ and $\overline{L_2} \in$ P-Sel, this also implies that P-Sel $\overline{\boldsymbol{\Delta}}$ P-Sel $\not\subseteq$ P-mc(3). $\square$

Note that Theorem 24 does not contradict Ogihara's result in [Ogi95] that $\Re^p_{2\text{-}tt}$(P-Sel) is contained in P-mc(2), since we consider the union and intersection of two possibly *different* sets in P-Sel, whereas the two queries in a $\leq^p_{2\text{-}tt}$-reduction are asked to the *same* set in P-Sel. Clearly, if P-Sel were closed under join, then we indeed would have a contradiction. However, P-Sel is not closed under join [HJ95].

Next, we prove that in terms of the levels of the GCH hierarchy, the class of intersections of P-selective sets can be clearly distinguished from, e.g., the class of unions of P-selective sets. This is in contrast with the P-mc hierarchy, which by the above theorem is not refined enough to sense this distinction. We note that some parts of this Theorem 25 extend Hemaspaandra and Jiang's results [HJ95], and also Rao's observation that P-Sel **op** P-Sel $\not\subseteq$ SH for any Boolean operation **op** chosen from $\{\wedge, \vee, \boldsymbol{\Delta}\}$ [Rao94]. Note further that Part 2 of Theorem 25 still leaves a gap between the upper and the lower bound for P-Sel $\wedge$ P-Sel.

Theorem 25. *1. For each $k \geq 2$,*
(a) $\oplus_k$(P-Sel) $\subseteq$ GC$(1, k, 1)$, *but* $\oplus_k$(P-Sel) $\not\subseteq$ SH $\cup$ GC$(1, k-1, 1)$, *and*
(b) $\vee_k$(P-Sel) $\subseteq$ GC$(1, k, 1)$, *but* $\vee_k$(P-Sel) $\not\subseteq$ SH $\cup$ GC$(1, k-1, 1)$.

2. P-Sel $\wedge$ P-Sel $\not\subseteq$ GC$(1,2,1)$, *but for each integer-valued* FP *function* $k(0^n)$ *satisfying* $1 \le k(0^n) \le n$, P-Sel $\wedge$ P-Sel $\subseteq$ GC$(\lceil \frac{n}{k(0^n)} \rceil, k(0^n), 1)$.
3. P-Sel **op** P-Sel $\not\subseteq$ Fair-GC$(1, n-1, 1)$ *for* **op** $\in \{\wedge, \boldsymbol{\Delta}, \overline{\boldsymbol{\Delta}}\}$.

Proof. 1. Let $L = \oplus_k(A_1, \ldots, A_k)$, where $A_i \in$ P-Sel via selector functions s_i for $i \in \{1, \ldots, k\}$. Let any inputs $x_1, \ldots, x_m$ be given, each having the form $\underline{i}a$ for some $i \in \{1, \ldots, k\}$ and $a \in \Sigma^*$. For each i, play a knock-out tournament among all strings a for which $\underline{i}a$ belongs to the inputs, where we say a_1 beats a_2 if $a_2 \preceq_{s_i} a_1$. Let $w_1, \ldots, w_k$ be the winners of the k tournaments. Define a GC$(1,k,1)$-selector for L to output $\{\underline{1}w_1, \ldots, \underline{k}w_k\}$. Clearly, at least one of these strings must be in L if at least one of the inputs is in L. The proof of $\vee_k$(P-Sel) $\subseteq$ GC$(1,k,1)$ is similar.

We only prove that P-Sel $\vee$ P-Sel $\not\subseteq$ SH by uniformly diagonalizing against all FP functions and all levels of SH. Define

$$L_1 \stackrel{\text{df}}{=} \bigcup_{\langle j,m\rangle\,:\, j\ge 0 \,\wedge\, m<\|R_{2j+1}\|} L_{1,\langle j,m\rangle} \quad \text{and} \quad L_2 \stackrel{\text{df}}{=} \bigcup_{\langle j,m\rangle\,:\, j\ge 0 \,\wedge\, m<\|R_{2j+1}\|} L_{2,\langle j,m\rangle},$$

where for each $j \ge 0$ and $m < \|R_{2j+1}\|$, the sets $L_{1,\langle j,m\rangle}$ and $L_{2,\langle j,m\rangle}$ are defined as follows:

$$\{i \in R_{2j+1} \mid i > f_j(r_{j,0}, \ldots, r_{j,m}) \wedge f_j(r_{j,0}, \ldots, r_{j,m}) \in \{r_{j,0}, \ldots, r_{j,m}\}\};$$

$$\{i \in R_{2j+1} \mid i < f_j(r_{j,0}, \ldots, r_{j,m}) \wedge f_j(r_{j,0}, \ldots, r_{j,m}) \in \{r_{j,0}, \ldots, r_{j,m}\}\}.$$

Clearly, $L_1 \in \mathcal{C}_1 \cap$ E and $L_2 \in \mathcal{C}_2 \cap$ E. Thus, by Lemma 23, $L_1, L_2 \in$ P-Sel. Assume P-Sel $\vee$ P-Sel $\subseteq$ SH, and in particular, $L_1 \cup L_2 \in$ S(m_0) via f_{j_0}. If $m_0 < \|R_{2j_0+1}\|$, then this contradicts the fact that $f_{j_0}(r_{j_0,0}, \ldots, r_{j_0,m_0})$ selects a string not in $L_1 \cup L_2$ though m_0 of the inputs are in $L_1 \cup L_2$. If $m_0 \ge \|R_{2j_0+1}\|$, then by our assumption that each transducer T_i appears infinitely often in the enumeration (see the remark after Lemma 23), there is an index j_1 such that $m_0 < \|R_{2j_1+1}\|$ and T_{j_1} computes f_{j_0}, and thus f_{j_0} is implicitly diagonalized against.

2. Let $k(0^n)$ be a function as in the theorem. Let $L = A \cap B$ for sets A and B, where $A \in$ P-Sel via f and $B \in$ P-Sel via g. We will define a GC$(\lceil \frac{n}{k(0^n)} \rceil, k(0^n), 1)$-selector s for L. Given n elements, rename them with respect to the linear order induced by f, i.e., we have $x_1 \preceq_f x_2 \preceq_f \cdots \preceq_f x_n$. Let $k \stackrel{\text{df}}{=} k(0^n)$. Now let h be the unique permutation of $\{1, \ldots, n\}$ such that for each $i, j \in \{1, \ldots, n\}$, $h(i) = j$ if and only if x_i is the jth element in the linear ordering of $\{x_1, \ldots, x_n\}$ induced by g. Partition the set $\{1, \ldots, n\}$ into k regions of at most $\lceil \frac{n}{k} \rceil$ elements:

$$R(l) \stackrel{\text{df}}{=} \left\{(l-1)\left\lceil\frac{n}{k}\right\rceil + 1, (l-1)\left\lceil\frac{n}{k}\right\rceil + 2, \ldots, l\left\lceil\frac{n}{k}\right\rceil\right\} \quad \text{for } 1 \le l \le k-1, \text{ and}$$

$$R(k) \stackrel{\text{df}}{=} \left\{(k-1)\left\lceil\frac{n}{k}\right\rceil + 1, (k-1)\left\lceil\frac{n}{k}\right\rceil + 2, \ldots, n\right\}.$$

Define $s(x_1, \ldots, x_n) \stackrel{\text{df}}{=} \{a_1, \ldots a_k\}$, where $a_l \stackrel{\text{df}}{=} x_{m(l)}$ and $m(l)$ is the $m \in R(l)$ such that $h(m)$ is maximum. Thus, for each region $R(l)$, a_l is the "most likely" element of its region to belong to B. Consider the permutation matrix of h with elements $(i, h(i))$, for $1 \leq i \leq n$. Let c_A be the "cutpoint" for A and let c_B be the "cutpoint" for B, i.e.,

$$\begin{array}{rcr} \{x_i \mid i < c_A\} \subseteq \overline{A} & \text{and} & \{x_i \mid i \geq c_A\} \subseteq A; \\ \{x_{h(i)} \mid h(i) < c_B\} \subseteq \overline{B} & \text{and} & \{x_{h(i)} \mid h(i) \geq c_B\} \subseteq B. \end{array}$$

Define

$$\begin{array}{ll} A_{\text{out}} \stackrel{\text{df}}{=} \{x_i \mid i < c_A\}; & A_{\text{in}} \stackrel{\text{df}}{=} \{x_i \mid i \geq c_A\}; \\ B_{\text{out}} \stackrel{\text{df}}{=} \{x_{h(i)} \mid h(i) < c_B\}; & B_{\text{in}} \stackrel{\text{df}}{=} \{x_{h(i)} \mid h(i) \geq c_B\}. \end{array}$$

Since $A_{\text{in}} \cap B_{\text{in}} \subseteq A \cap B$, it remains to show that at least one of the outputs a_l of s is in $A_{\text{in}} \cap B_{\text{in}}$, if the promise $\|\{x_1, \ldots, x_n\} \cap L\| \geq \lceil \frac{n}{k} \rceil$ is met. First observe that for each l, if $i \geq c_A$ holds for each $i \in R(l)$ and $R(l)$ contains an index i_0 such that $h(i_0) \geq c_B$, then $a_l \in A_{\text{in}} \cap B_{\text{in}}$. On the other hand, if c_A "cuts" a region $R(l_0)$, then in the worst case we have $a_{l_0} = (l_0 - 1)\lceil \frac{n}{k} \rceil + 1$ and $c_A = (l_0 - 1)\lceil \frac{n}{k} \rceil + 2$, and thus $a_{l_0} \notin A_{\text{in}}$ and at most $\lceil \frac{n}{k} \rceil - 1$ elements of A_{in} can have an index in $R(l_0)$. However, if $\|\{x_1, \ldots, x_n\} \cap L\| \geq \lceil \frac{n}{k} \rceil$, then there must exist an l_1 with $l_1 > l_0$ such that for each $i \in R(l_1)$ it holds that $i \geq c_A$, and thus, $a_{l_1} \in A_{\text{in}} \cap B_{\text{in}}$. This proves $L \in \mathrm{GC}(\lceil \frac{n}{k} \rceil, k, 1)$ via s.

The proof of P-Sel $\wedge$ P-Sel $\not\subseteq \mathrm{GC}(1, 2, 1)$ is similar as in Part 3.

3. We only prove P-Sel $\wedge$ P-Sel $\not\subseteq$ Fair-GC$(1, n-1, 1)$ (the other cases are similar). Define

$$L_1 \stackrel{\text{df}}{=} \left\{ i \;\middle|\; \begin{array}{l} (\exists j \geq 0)\, [i \in R_{2j+1} \text{ and } i \geq w_j \text{ for the smallest string} \\ w_j \in R_{2j+1} \text{ such that } f_j(R_{2j+1}) \subseteq R_{2j+1} - \{w_j\}] \end{array} \right\};$$

$$L_2 \stackrel{\text{df}}{=} \left\{ i \;\middle|\; \begin{array}{l} (\exists j \geq 0)\, [i \in R_{2j+1} \text{ and } i \leq w_j \text{ for the smallest string} \\ w_j \in R_{2j+1} \text{ such that } f_j(R_{2j+1}) \subseteq R_{2j+1} - \{w_j\}] \end{array} \right\}.$$

As before, $L_1, L_2 \in$ P-Sel. Assume there is a Fair-GC$(1, n-1, 1)$-selector f_{j_0} for $L_1 \cap L_2$. First observe that the "fairness condition" is satisfied if f_{j_0} has all strings from R_{2j_0+1} as inputs, since $\|R_{2j_0+1}\| = 2^{2^{\mu(2j_0+1)}} - \mu(2j_0 + 1)$ and the length of the largest string in R_{2j_0+1} is at most $2^{\mu(2j_0+1)}$. For the Fair-GC$(1, n-1, 1)$-selector f_{j_0}, there must exist a smallest string $w_{j_0} \in R_{2j_0+1}$ such that $f_{j_0}(R_{2j_0+1})$ is contained in $R_{2j_0+1} - \{w_{j_0}\}$, and thus, $\{w_{j_0}\} = L_1 \cap L_2 \cap R_{2j_0+1}$. This would contradict $f_{j_0}(R_{2j_0+1})$ not selecting w_{j_0}. □

Statement 2 of the above theorem immediately gives the first part of Corollary 26. Note that, even though this $\mathrm{GC}(\sqrt{n}, \sqrt{n}, 1)$ upper bound on P-Sel $\wedge$ P-Sel may not be strong enough to prove the second part of the corollary, the proof of this second part does easily follow from the P-Sel $\wedge$ P-Sel $\subseteq$ P-mc(3) result of Theorem 24 via Ogihara's result that the assumption NP $\subseteq$ P-mc(3) implies the collapse of P = NP [Ogi95].

Corollary 26. *1.* P-Sel $\wedge$ P-Sel $\subseteq \mathrm{GC}(\sqrt{n}, \sqrt{n}, 1)$.
2. NP $\subseteq$ P-Sel $\wedge$ P-Sel $\implies$ P = NP.

4.2 The Structure of the GC Hierarchy

In this subsection, we study the internal structure of GCH. We start with determining for which parameters b, c, and d the class $\mathrm{GC}(b,c,d)$ is nontrivial (i.e., satisfies $\mathrm{GC}(b,c,d) \neq \mathcal{P}(\Sigma^*)$, yet contains not only finite sets). Recall that $w_{i,1},\ldots,w_{i,s}$ are the lexicographically smallest s length $e(i)$ strings, for $i \geq 0$ and $s \leq 2^{e(i)}$ (the function $e(i)$ is defined in Section 3). The proofs of some of the more technical lemmas in this subsection are deferred to Section 4.3. For instance, the proof of Lemma 27 below can be found in Section 4.3.

Lemma 27. *Let $b,c,d \in \mathbb{N}^+$ with $d \leq c$ and $d \leq b$. Then,*

1. $(\exists A)\,[A \in \mathrm{GC}(b,c,d) \ \wedge\ \|A\| = \infty]$, *and*
2. $(\exists B)\,[B \notin \mathrm{GC}(b,c,d) \ \wedge\ \|B\| = \infty]$.

Theorem 28. *Let $b,c,d \in \mathbb{N}^+$.*

1. *Every set in $\mathrm{GC}(b,c,d)$ is finite if and only if $d > b$ or $d > c$.*
2. *If $d \leq b$ and $d \leq c$, then $\mathrm{GC}(b,c,d)$ is nontrivial.*

Proof. If $d > c$ or $d > b$, then by Definition 20, every set in $\mathrm{GC}(b,c,d)$ is finite. On the other hand, if $d \leq b$ and $d \leq c$, then by Lemma 27.1, there is an infinite set in $\mathrm{GC}(b,c,d)$. Hence, every set in $\mathrm{GC}(b,c,d)$ is finite if and only if $d > b$ or $d > c$. Furthermore, if $d \leq b$ and $d \leq c$, then $\mathrm{GC}(b,c,d) \neq \mathcal{P}(\Sigma^*)$ by Lemma 27.2. □

Now we turn to the relationships between the nontrivial classes within GCH. Given any parameters b,c,d and i,j,k, we seek to determine which of $\mathrm{GC}(b,c,d)$ and $\mathrm{GC}(i,j,k)$ is contained in the other class (and if this inclusion is strict), or whether they are mutually incomparable. For classes $\mathcal{A}$ and $\mathcal{B}$, let $\mathcal{A} \bowtie \mathcal{B}$ denote that $\mathcal{A}$ and $\mathcal{B}$ are incomparable, i.e., $\mathcal{A} \not\subseteq \mathcal{B}$ and $\mathcal{B} \not\subseteq \mathcal{A}$. Theorem 31 will establish these relations for almost all the cases and is proven by making extensive use of the Inclusion Lemma and the Diagonalization Lemma below. The proofs of Lemmas 29 and 30 can be found in Section 4.3.

Lemma 29 (Inclusion Lemma). Let $b,c,d \in \mathbb{N}^+$ and $l,m,n \in \mathbb{N}$ be given such that each GC class below is nontrivial. Then,

1. $\mathrm{GC}(b,c,c) = \mathrm{S}(b,c)$.
2. $\mathrm{GC}(b,c,d+n) \subseteq \mathrm{GC}(b+l,c+m,d)$.
3. If $l \geq n$ and $m \geq n$, then $\mathrm{GC}(b,c,c) \subseteq \mathrm{GC}(b+l,c+m,c+n)$.
4. If $l \leq n$ and $m \leq n$, then $\mathrm{GC}(b+l,c+m,d+n) \subseteq \mathrm{GC}(b,c,d)$.

Lemma 30 (Diagonalization Lemma). Let $b,c,d \in \mathbb{N}^+$ and $l,m,n,q \in \mathbb{N}$ be given such that each GC class below is nontrivial. Then,

1. If $l \geq n+1$, then $(\exists L)\,[L \in \mathrm{GC}(b+l,c+m,d+n) - \mathrm{GC}(b,c+q,d)]$.
2. If $m \geq n+1$, then $(\exists L)\,[L \in \mathrm{GC}(b+l,c+m,d+n) - \mathrm{GC}(b+q,c,d)]$.
3. If $(n \geq l+1$ or $n \geq m+1)$, then $(\exists L)\,[L \in \mathrm{GC}(b,c,d) - \mathrm{GC}(b+l,c+m,d+n)]$.

Theorem 31. *Let $b,c,d \in \mathbb{N}^+$ and $i,j,k \in \mathbb{N}$ be given such that each GC class below is nontrivial. Then,*

1. $\mathrm{GC}(b,c,d+k) \subset \mathrm{GC}(b+i,c+j,d)$ *if* $i \geq 1$ *or* $j \geq 1$ *or* $k \geq 1$.
2. $\mathrm{GC}(b,c+j,d+k) \subset \mathrm{GC}(b+i,c,d)$ *if* $1 \leq j \leq k$.
3. $\mathrm{GC}(b,c+j,d+k) \bowtie \mathrm{GC}(b+i,c,d)$ *if* $j > k \geq 1$.
4. $\mathrm{GC}(b+i,c,d+k) \subset \mathrm{GC}(b,c+j,d)$ *if* $1 \leq i \leq k$.
5. $\mathrm{GC}(b+i,c,d+k) \bowtie \mathrm{GC}(b,c+j,d)$ *if* $i > k \geq 1$.
6. $\mathrm{GC}(b+i,c,d) \bowtie \mathrm{GC}(b,c+j,d)$ *if* $i \geq 1$ *and* $j \geq 1$.
7. $\mathrm{GC}(b+i,c+j,d+k) \subset \mathrm{GC}(b,c,d)$ *if* ($1 \leq i < k$ *and* $1 \leq j \leq k$) *or* ($1 \leq j < k$ *and* $1 \leq i \leq k$).
8. $\mathrm{GC}(b+i,c+j,d+k) = \mathrm{GC}(b,c,d)$ *if* $i = j = k$ *and* $c = d$.
9. $\mathrm{GC}(b+i,c+j,d+k) \bowtie \mathrm{GC}(b,c,d)$ *if* $1 \leq i < k < j$ *or* $1 \leq j < k < i$.

Proof. The proof is done by repeatedly applying Lemma 29 and Lemma 30. Unless otherwise specified, l, m, and n in the lemmas correspond to i, j, and k in this proof.

1. The inclusion is clear (see Lemma 29.2). For the strictness of the inclusion, we have to consider three cases. If $i \geq 1$, then by Lemma 30.1 with $n = q = 0$, there exists a set $L \in \mathrm{GC}(b+i,c+j,d) - \mathrm{GC}(b,c,d)$. By Lemma 29.2 with $l = m = 0$, $L \notin \mathrm{GC}(b,c,d+k)$. The case of $j \geq 1$ is treated similar, using Lemma 30.2 instead of Lemma 30.1. Finally, if $k \geq 1$, then by Lemma 30.3 with $l = m = 0$, we have $L \in \mathrm{GC}(b,c,d) - \mathrm{GC}(b,c,d+k)$. By Lemma 29.2 with $n = 0$, $L \in \mathrm{GC}(b+i,c+j,d)$.

2. Applying Lemma 29.4 with $l = 0$ and then Lemma 29.2 with $m = n = 0$, we have $\mathrm{GC}(b,c+j,d+k) \subseteq \mathrm{GC}(b,c,d) \subseteq \mathrm{GC}(b+i,c,d)$. By Lemma 30.3 with $l = 0$ (i.e., $n \geq 1$), there exists a set $L \in \mathrm{GC}(b,c,d) - \mathrm{GC}(b,c+j,d+k)$. By Lemma 29.2 with $m = n = 0$, $L \in \mathrm{GC}(b+i,c,d)$.

3. "$\not\subseteq$" follows from Lemma 30.2 with $q = i$ and $l = 0$. "$\not\supseteq$" follows as in Part 2.

4. Applying Lemma 29.4 with $m = 0$ and then Lemma 29.2 with $l = n = 0$, we have $\mathrm{GC}(b+i,c,d+k) \subseteq \mathrm{GC}(b,c,d) \subseteq \mathrm{GC}(b,c+m,d)$. The strictness of the inclusion follows as in Part 2, where Lemma 30.3 is applied with $m = 0$ instead of $l = 0$.

5. "$\not\subseteq$" follows from Lemma 30.1 with $q = j$ and $m = 0$. "$\not\supseteq$" holds by Lemma 30.3 with $m = 0$ (i.e., $n \geq 1$) and Lemma 29.2 with $l = n = 0$.

6. "$\not\subseteq$" holds, as by Lemma 30.1 with $q = j$ and $m = n = 0$, there exists a set L in $\mathrm{GC}(b+i,c,d) - \mathrm{GC}(b,c+j,d)$. "$\not\supseteq$" similarly follows from Lemma 30.2 with $q = i$ and $l = n = 0$.

7. By Lemma 29.4, $\mathrm{GC}(b+i,c+j,d+k) \subseteq \mathrm{GC}(b,c,d)$. By Lemma 30.3, if $n > l$ or $n > m$, then there exists a set $L \in \mathrm{GC}(b,c,d) - \mathrm{GC}(b+i,c+j,d+k)$.

8. The equality follows from Lemma 29.3 and Lemma 29.4.

9. Let $i < k < j$. Then, by Lemma 30.2 with $q = 0$, there exists a set L in $\mathrm{GC}(b+i,c+j,d+k) - \mathrm{GC}(b,c,d)$. Conversely, by Lemma 30.3, there exists a set L in $\mathrm{GC}(b,c,d) - \mathrm{GC}(b+i,c+j,d+k)$. If $j < k < i$, the incomparability of $\mathrm{GC}(b,c,d)$ and $\mathrm{GC}(b+i,c+j,d+k)$ similarly follows from Lemma 30.1 and Lemma 30.3. □

Note that Theorem 31 does not settle all possible relations between the GC classes. That is, the relation between $\mathrm{GC}(b,c,d)$ and $\mathrm{GC}(b+i,c+j,d+k)$ is left open for the case of ($k \leq i$ and $k \leq j$ and $c \neq d$). Figure 3 shows the relations amongst all nontrivial classes $\mathrm{GC}(b,c,d)$ with $1 \leq b,c,d \leq 3$, as they are proven in Theorem 31 and Theorem 32 (those relations not established by Theorem 31 are marked by "$*$" in Figure 3 and are proven separately as Theorem 32 below). For instance, $\mathrm{S}(2) = \mathrm{GC}(3,2,2) \subset \mathrm{GC}(3,3,2)$ holds by the first part of Theorem 31 with $b = 3$, $c = d = 2$, $i = k = 0$, and $j = 1$. The "A" in Figure 3 indicates that, while the inclusion holds by Lemma 29.4, the strictness of the inclusion was observed by A. Nickelsen and appears here with his kind permission.

Theorem 32. *1.* [Nic94] $\mathrm{GC}(2,3,2) \subset \mathrm{GC}(1,2,1)$.
2. $\mathrm{GC}(3,3,2) \bowtie \mathrm{GC}(1,2,1)$.
3. $\mathrm{GC}(3,3,2) \subset \mathrm{GC}(2,2,1)$.

Proof. Both the inclusion $\mathrm{GC}(2,3,2) \subseteq \mathrm{GC}(1,2,1)$ and the inclusion $\mathrm{GC}(3,3,2) \subseteq \mathrm{GC}(2,2,1)$ follow from Lemma 29.4 with $l = m = n = 1$. We now provide those diagonalizations required to complete the proof of the theorem.

1. For proving $\mathrm{GC}(1,2,1) \not\subseteq \mathrm{GC}(2,3,2)$, we will define a set $L = \bigcup_{i\geq 1} L_i$ such that for each i, $L_i \subseteq W_{i,4}$, and if $f_i(W_{i,4}) \subseteq W_{i,4}$ and $\|f_i(W_{i,4})\| = 3$, then we make sure that $\|L_i\| = 2$ and $\|L_i \cap f_i(W_{i,4})\| = 1$. This ensures that for no $i \geq 1$ can f_i be a $\mathrm{GC}(2,3,2)$-selector for L. For example, this can be accomplished by defining L_i as follows:

$$
\begin{aligned}
\chi_L(w_{i,1},\ldots,w_{i,4}) &= 0101 \text{ if } f_i(W_{i,4}) = \{w_{i,1},w_{i,2},w_{i,3}\},\\
\chi_L(w_{i,1},\ldots,w_{i,4}) &= 1010 \text{ if } f_i(W_{i,4}) = \{w_{i,1},w_{i,2},w_{i,4}\},\\
\chi_L(w_{i,1},\ldots,w_{i,4}) &= 1100 \text{ if } f_i(W_{i,4}) = \{w_{i,1},w_{i,3},w_{i,4}\},\\
\chi_L(w_{i,1},\ldots,w_{i,4}) &= 1100 \text{ if } f_i(W_{i,4}) = \{w_{i,2},w_{i,3},w_{i,4}\}.
\end{aligned}
$$

Note that if $f_i(W_{i,4})$ outputs a string not in $W_{i,4}$ or the number of output strings is different from 3, then (by Definition 20 and the remark following Definition 20) f_i immediately is disqualified from being a $\mathrm{GC}(2,3,2)$-selector for L (and we set $L_i = \emptyset$ in this case). Thus, $L \notin \mathrm{GC}(2,3,2)$. On the other hand, $L \in \mathrm{GC}(1,2,1)$ can be seen as follows: Given any set of inputs X with $\|X\| \geq 2$, we can w.l.o.g. assume that $X \subseteq \bigcup_{i\geq 1} W_{i,4}$; since smaller strings can be solved by brute force, we may even assume that $X \subseteq W_{j,4}$ for some j. Suppose further that $\|L \cap X\| \geq 1$. Define $g(X) \stackrel{\mathrm{df}}{=} X$ if $\|X\| = 2$; and if $\|X\| > 2$, define $g(X)$ to output $\{w_{j,1},w_{j,4}\}$ if $\{w_{j,1},w_{j,4}\} \subseteq X$, and to output $\{w_{j,2},w_{j,3}\}$ otherwise. Since $\|L \cap \{w_{j,1},w_{j,4}\}\| = 1$ and $\|L \cap \{w_{j,2},w_{j,3}\}\| = 1$ holds in each of the four cases above, it follows that $\|L \cap g(X)\| \geq 1$. Hence, $L \in \mathrm{GC}(1,2,1)$ via g.

2. For proving $\mathrm{GC}(1,2,1) \not\subseteq \mathrm{GC}(3,3,2)$, L is defined as $\bigcup_{i\geq 1} L_i$, where $L_i \subseteq W_{i,5}$, and if $f_i(W_{i,5}) \subseteq W_{i,5}$ and $\|f_i(W_{i,5})\| = 3$, then we make sure that $\|L_i\| = 3$ and $\|L_i \cap f_i(W_{i,5})\| = 1$. This ensures that for no $i \geq 1$ can f_i be a $\mathrm{GC}(3,3,2)$-selector for L. For example, this can be achieved by defining L_i as follows:

$$
\begin{aligned}
\chi_L(w_{i,1},\ldots,w_{i,5}) &= 01011 \text{ if } f_i(W_{i,5}) = \{w_{i,1},w_{i,2},w_{i,3}\},\\
\chi_L(w_{i,1},\ldots,w_{i,5}) &= 10101 \text{ if } f_i(W_{i,5}) = \{w_{i,1},w_{i,2},w_{i,4}\},\\
\chi_L(w_{i,1},\ldots,w_{i,5}) &= 10110 \text{ if } f_i(W_{i,5}) = \{w_{i,1},w_{i,2},w_{i,5}\},\\
\chi_L(w_{i,1},\ldots,w_{i,5}) &= 01101 \text{ if } f_i(W_{i,5}) = \{w_{i,1},w_{i,3},w_{i,4}\},
\end{aligned}
$$

$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 01011 \text{ if } f_i(W_{i,5}) = \{w_{i,1}, w_{i,3}, w_{i,5}\},$$
$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 01101 \text{ if } f_i(W_{i,5}) = \{w_{i,1}, w_{i,4}, w_{i,5}\},$$
$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 10101 \text{ if } f_i(W_{i,5}) = \{w_{i,2}, w_{i,3}, w_{i,4}\},$$
$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 11010 \text{ if } f_i(W_{i,5}) = \{w_{i,2}, w_{i,3}, w_{i,5}\},$$
$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 10110 \text{ if } f_i(W_{i,5}) = \{w_{i,2}, w_{i,4}, w_{i,5}\},$$
$$\chi_L(w_{i,1},\ldots,w_{i,5}) = 11010 \text{ if } f_i(W_{i,5}) = \{w_{i,3}, w_{i,4}, w_{i,5}\}.$$

As argued above, this shows that $L \notin \mathrm{GC}(3,3,2)$. For proving that L is in $\mathrm{GC}(1,2,1)$, let a set X of inputs be given and suppose w.l.o.g. that $\|X\| \geq 3$ and $X \subseteq W_{j,5}$ for some j. Note that for each choice of X, at least one of $\{w_{j,1}, w_{j,2}\}$, $\{w_{j,2}, w_{j,3}\}$, $\{w_{j,3}, w_{j,4}\}$, $\{w_{j,4}, w_{j,5}\}$, or $\{w_{j,5}, w_{j,1}\}$ must be contained in X. On the other hand, each of $\{w_{j,1}, w_{j,2}\}$, $\{w_{j,2}, w_{j,3}\}$, $\{w_{j,3}, w_{j,4}\}$, $\{w_{j,4}, w_{j,5}\}$, and $\{w_{j,5}, w_{j,1}\}$ has (by construction of L) at least one string in common with L_j if L_j is not set to the empty set. From these comments the action of the $\mathrm{GC}(1,2,1)$-selector is clear.

For proving $\mathrm{GC}(3,3,2) \not\subseteq \mathrm{GC}(1,2,1)$, define a set $L \subseteq \bigcup_{i\geq 1} W_{i,3}$ as follows:

$$\chi_L(w_{i,1}, w_{i,2}, w_{i,3}) = 100 \text{ if } f_i(W_{i,3}) = \{w_{i,2}, w_{i,3}\},$$
$$\chi_L(w_{i,1}, w_{i,2}, w_{i,3}) = 010 \text{ if } f_i(W_{i,3}) = \{w_{i,1}, w_{i,3}\},$$
$$\chi_L(w_{i,1}, w_{i,2}, w_{i,3}) = 001 \text{ if } f_i(W_{i,3}) = \{w_{i,1}, w_{i,2}\}.$$

Since in each case $\|L \cap W_{i,3}\| = 1$ but $L \cap f_i(W_{i,3}) = \emptyset$, L cannot be in $\mathrm{GC}(1,2,1)$. On the other hand, L is easily seen to be in $\mathrm{GC}(3,3,2)$ via a selector that first solves all "small" inputs (i.e., those strings not of maximum length) by brute force and then outputs two small members of L (and one arbitrary input) if those can be found, or three arbitrary inputs if no more than one small member of L is found by brute force. Note that the $\mathrm{GC}(3,3,2)$-promise is not satisfied in the latter case.

Part 3 follows from Part 2, as $\mathrm{GC}(1,2,1) \subset \mathrm{GC}(2,2,1)$. □

4.3 Some Proofs Deferred from Section 4.2

Proof of Lemma 27. 1. Let $A = \Sigma^*$. Given n distinct strings $y_1, \ldots, y_n$, define

$$f(y_1,\ldots,y_n) \stackrel{\mathrm{df}}{=} \begin{cases} \{y_1,\ldots,y_c\} & \text{if } n \geq c \\ \{y_1,\ldots,y_n\} & \text{if } n < c. \end{cases}$$

Clearly, $f \in \mathrm{FP}$, $f(y_1,\ldots,y_n) \subseteq A$, and $\|f(y_1,\ldots,y_n)\| \leq c$.

If $\|\{y_1,\ldots,y_n\} \cap A\| \geq b$, then $n \geq b$, and thus we have

$$\|f(y_1,\ldots,y_n) \cap A\| = c \geq d$$

if $n \geq c$, and if $n < c$, then

$$\|f(y_1,\ldots,y_n) \cap A\| = n \geq b \geq d.$$

By Definition 20, $A \in \mathrm{GC}(b,c,d)$.

2. We will define $B \stackrel{\text{df}}{=} \bigcup_{i\geq 1} B_i$ such that for no i with $b+c-d+1 \leq 2^{e(i)}$ can f_i be a $\mathrm{GC}(b,c,d)$-selector for B. By our assumption about the enumeration of FP functions (recall the remark after Lemma 23), this suffices. For each i with

$$b+c-d+1 > 2^{e(i)},$$

set $B_i \stackrel{\text{df}}{=} \emptyset$. For each i such that

$$b+c-d+1 \leq 2^{e(i)},$$

let F_i and W_i be shorthands for the sets $f_i(w_{i,1},\ldots,w_{i,b+c-d+1})$ and $\{w_{i,1},\ldots,w_{i,b+c-d+1}\}$, respectively, and let $w_{i,j_1},\ldots,w_{i,j_{d-1}}$ be the first $d-1$ strings in F_i (if $\|F_i\| \geq d$). W.l.o.g., assume $F_i \subseteq W_i$ and $\|F_i\| \leq c$ (if not, f_i automatically is disqualified from being a $\mathrm{GC}(b,c,d)$-selector).

Define

$$B_i \stackrel{\text{df}}{=} \begin{cases} \{w_{i,j_1},\ldots,w_{i,j_{d-1}}\} \cup (W_i - F_i) & \text{if } d \leq \|F_i\| \\ W_i & \text{if } d > \|F_i\|. \end{cases}$$

Thus, either we have

$$\|W_i \cap B\| \geq (d-1) + ((b+c-d+1)-c) = b \text{ and } \|F_i \cap B\| < d,$$

or we have

$$\|W_i \cap B\| = b+c-d+1 > b \text{ and } \|F_i \cap B\| < d.$$

Hence, $B \notin \mathrm{GC}(b,c,d)$. □

Proof of Lemma 29. 1. & 2. Immediate from the definitions of GC and S classes.

3. Let $l \geq n$ and $m \geq n$. By Parts 1 and 2 of this lemma and by Theorem 5, we have

$$\begin{aligned} \mathrm{GC}(b,c,c) = \mathrm{S}(b,c) \;&=\; \mathrm{S}(b+n,c+n) \;=\; \mathrm{GC}(b+n,c+n,c+n) \\ &\subseteq \mathrm{GC}(b+l,c+m,c+n). \end{aligned}$$

4. Suppose $m \leq l \leq n$ and $L \in \mathrm{GC}(b+l,c+m,d+n)$ via $f \in \mathrm{FP}$. As in the proof of Theorem 5, let finitely many strings $z_1,\ldots,z_{b+2l-1}$, each belonging to L, be hard-coded into the transducer computing function g defined below. Given inputs $Y = \{y_1,\ldots,y_t\}$, choose (if possible) l strings $z_{i_1},\ldots,z_{i_l} \notin Y$, and define

$$g(Y) \stackrel{\text{df}}{=} \begin{cases} f(Y \cup \{z_{i_1},\ldots,z_{i_l}\}) - \{u_1,\ldots,u_l\} & \text{if } z_{i_1},\ldots,z_{i_l} \notin Y \text{ exist} \\ f(Y) - \{v_1,\ldots,v_m\} & \text{otherwise,} \end{cases}$$

where $\{u_1,\ldots,u_l\}$ contains *all* z-strings output by f, say there are h with $h \leq l$, the remaining $l-h$ u-strings are arbitrary y-strings of the output of f, and similarly, $v_1,\ldots,v_m$ are arbitrary output strings of f. Clearly, $g \in \mathrm{FP}$ and $g(Y) \subseteq Y$. Moreover, $\|g(Y)\| \leq c+m-l \leq c$ if $z_{i_1},\ldots,z_{i_l} \notin Y$ exist; otherwise, we trivially have $\|g(Y)\| \leq c$. Note that if $z_{i_1},\ldots,z_{i_l} \notin Y$ do not exist, then

$$\|Y \cap \{z_1,\ldots,z_{b+2l-1}\}\| \geq b+l.$$

Thus, if $\|L \cap Y\| \geq b$, then either $\|L \cap (Y \cup \{z_{i_1}, \ldots, z_{i_l}\})\| \geq b + l$ implies

$$\|L \cap g(Y)\| \geq d + n - l \geq d,$$

or $\|L \cap Y\| \geq b + l$ implies

$$\|L \cap g(Y)\| \geq d + n - m \geq d.$$

This establishes that $m \leq l \leq n$ implies

$$\mathrm{GC}(b + l, c + m, d + n) \subseteq \mathrm{GC}(b, c, d).$$

By symmetry, we similarly obtain that $l \leq m \leq n$ implies the containment of $\mathrm{GC}(b + l, c + m, d + n)$ in $\mathrm{GC}(b, c, d)$, if we exchange l and m in the above argument. Since ($m \leq l \leq n$ or $l \leq m \leq n$) if and only if ($l \leq n$ and $m \leq n$), the proof is complete. □

Proof of Lemma 30. 1. The diagonalization part of the proof is analogous to the proof of Lemma 27.2, the only difference being that here we have $c + q$ instead of c. Also, it will be useful to require that any (potential) selector f_i for some set in $\mathrm{GC}(b, c + q, d)$ has the property that for any set of inputs W with $\|W\| \geq c + q$, $\|f_i(W)\|$ is *exactly* $c + q$. By the remark after Definition 20, this results in an equivalent definition of the GC class and can w.l.o.g. be assumed. The construction of set $L = \bigcup_{i \geq 1} L_i$ is as follows. For each i with

$$2^{e(i)} < b + c + q - d + 1,$$

set $L_i \stackrel{\mathrm{df}}{=} \emptyset$. For each i such that

$$2^{e(i)} \geq b + c - d + 1,$$

let F_i and W_i be shorthands for the sets $f_i(w_{i,1}, \ldots, w_{i,b+c+q-d+1})$ and $\{w_{i,1}, \ldots, w_{i,b+c+q-d+1}\}$, respectively, and let $w_{i,j_1}, \ldots, w_{i,j_{d-1}}$ be the first $d - 1$ strings in F_i (if $\|F_i\| \geq d$).

If $\|F_i\| = c + q\,(\geq d)$ and $F_i \subseteq W_i$, then set

$$L_i \stackrel{\mathrm{df}}{=} \{w_{i,j_1}, \ldots, w_{i,j_{d-1}}\} \cup (W_i - F_i);$$

otherwise, set $L_i \stackrel{\mathrm{df}}{=} W_i$. By the argument given in the proof of Lemma 27.2, $L \notin \mathrm{GC}(b, c + q, d)$.

Now we prove that $L \in \mathrm{GC}(b + l, c + m, d + n)$ if $l > n$. Given any distinct input strings $y_1, \ldots, y_t$, suppose they are lexicographically ordered (i.e., $y_1 <_{\mathrm{lex}} \cdots <_{\mathrm{lex}} y_t$), each y_s is in W_j for some j, and $y_k <_{\mathrm{lex}} \cdots <_{\mathrm{lex}} y_t$ are all strings of maximum length for some k with $1 \leq k \leq t$. Define a $\mathrm{GC}(b + l, c + m, d + n)$-selector f for L as follows:

1. For $i \in \{1, \ldots, k - 1\}$, decide by brute force whether y_i is in L. Let v denote $\|\{y_1, \ldots, y_{k-1}\} \cap L\|$. Output $\min\{v, d + n\}$ strings in L. If $v \geq d + n$ then halt, otherwise go to 2.
2. If $t \geq k + (d + n - v) - 1$, then output $y_k, \ldots, y_{k+(d+n-v)-1}$; otherwise, output $y_1, \ldots, y_t$.

Clearly, $f \in \mathrm{FP}$, $f(y_1, \ldots, y_t) \subseteq \{y_1, \ldots, y_t\}$, and since $\mathrm{GC}(b+l, c+m, d+n)$ is nontrivial, we have:

$$\|f(y_1, \ldots, y_t)\| \leq v + (d+n-v) \leq c+m.$$

Now we prove that

$$\|\{y_1, \ldots, y_t\} \cap L\| \geq b+l \implies \|f(y_1, \ldots, y_t) \cap L\| \geq d+n.$$

Let i be such that $e(i)$ is the length of $y_k, \ldots, y_t$. Clearly, if $\|F_i\| \neq c+q$, then by construction of L and f, either f outputs $d+n$ strings in L, or

$$L \cap \{y_1, \ldots, y_t\} = f(y_1, \ldots, y_t),$$

and so we are done. Similarly, if f halts in Step 1 because of $v \geq d+n$, then we are also done.

So suppose $v < d+n$, $\|\{y_1, \ldots, y_t\} \cap L\| \geq b+l$, and $\|F_i\| = c+q \geq d$. Recall that $w_{i,j_{d-1}}$ is the $(d-1)$st string in F_i. Define

$$D \stackrel{\mathrm{df}}{=} \{y_k, \ldots, y_t\} \cap \{w_{i,1}, \ldots, w_{i,j_{d-1}}\}.$$

By construction of L, we have $\{w_{i,1}, \ldots, w_{i,j_{d-1}}\} \subseteq L$, so $D \subseteq L$. That is,

$$\{y_k, \ldots, y_{k+\|D\|-1}\} \subseteq L. \tag{1}$$

Since $\|\{y_k, \ldots, y_t\} \cap L\| \geq b+l-v$, we have

$$t-(k-1) \geq b+l-v \geq d+n-v,$$

and thus,

$$t \geq k+(d+n-v)-1.$$

This implies:

$$\{y_k, \ldots, y_{k+(d+n-v)-1}\} \subseteq f(y_1, \ldots, y_t). \tag{2}$$

Thus, if $d+n-v \leq \|D\|$, we obtain from (1) that $\{y_k, \ldots, y_{k+(d+n-v)-1}\} \subseteq L$, which in turn implies with (2) that

$$\|L \cap f(y_1, \ldots, y_t)\| \geq v + (d+n-v) = d+n.$$

So it remains to show that $d+n-v \leq \|D\|$. Observe that

$$b+l \leq \|\{y_1, \ldots, y_t\} \cap L\| \leq v + \|D\| + b - d + 1,$$

since $\|W_i - F_i\| = (b+c+q-d+1) - (c+q) = b-d+1$ (here we need that $\|F_i\| = c+q$ rather than $\|F_i\| \leq c+q$ for f_i to be a $\mathrm{GC}(b, c+q, d)$-selector). Thus, $v + \|D\| + b - d + 1 \geq b+l$. By the assumption that $l \geq n+1$, we obtain $d+n-v \leq \|D\|$.

Parts 2 and 3 of this theorem can be proven by similar arguments. □

Acknowledgments

We thank Hans-Jörg Burtschick, Johannes Köbler, and Arfst Nickelsen for interesting discussions. In particular, we are indebted to Arfst Nickelsen for his kind permission to include his proof of the first item in Theorem 32 and to Johannes Köbler for pointing out to us an error in an earlier version of this paper.

The first author was supported in part by grants NSF-CCR-8957604, NSF-INT-9116781/JSPS-ENG-207, NSF-INT-9513368/DAAD-315-PRO-fo-ab, and NSF-CCR-9322513. The second author was supported in part by a postdoctoral fellowship from the Chinese Academy of Sciences, and by grant NSF-CCR-8957604. The third author was supported in part by a DAAD research visit grant, and grants NSF-CCR-9322513, NSF-CCR-8957604, and NSF-INT-9513368/DAAD-315-PRO-fo-ab. The fourth author was supported in part by grant NSF-INT-9116781/JSPS-ENG-207.

This work was done in part while the first author was visiting Friedrich-Schiller-Universität Jena and the Tokyo Institute of Technology, and in part while the other three authors were visiting the University of Rochester. The second author's current address is: Suite 1100, 123 Front Street, Toronto, Ontario, M5J 2M3 Canada.

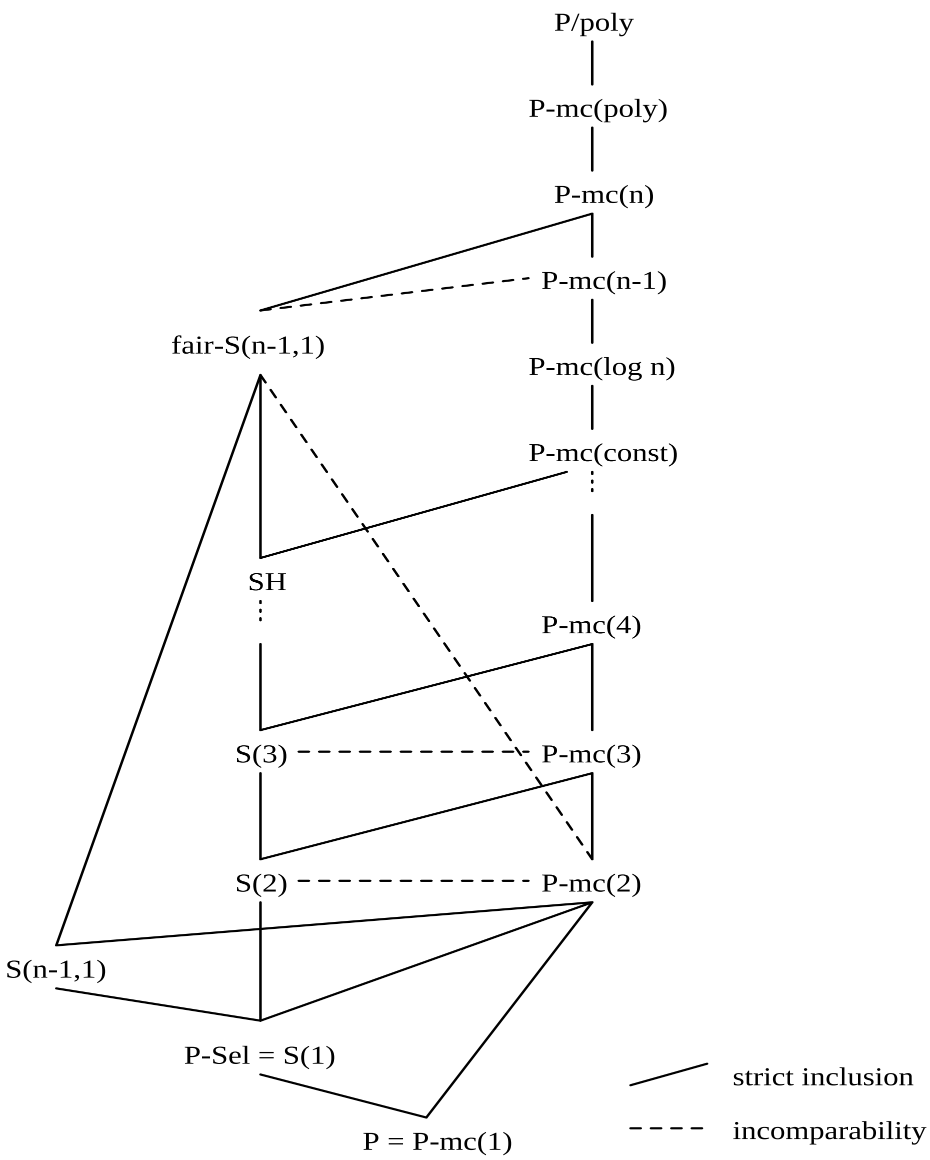

Figure 2: Inclusion relationships among S, Fair-S, and P-mc classes.

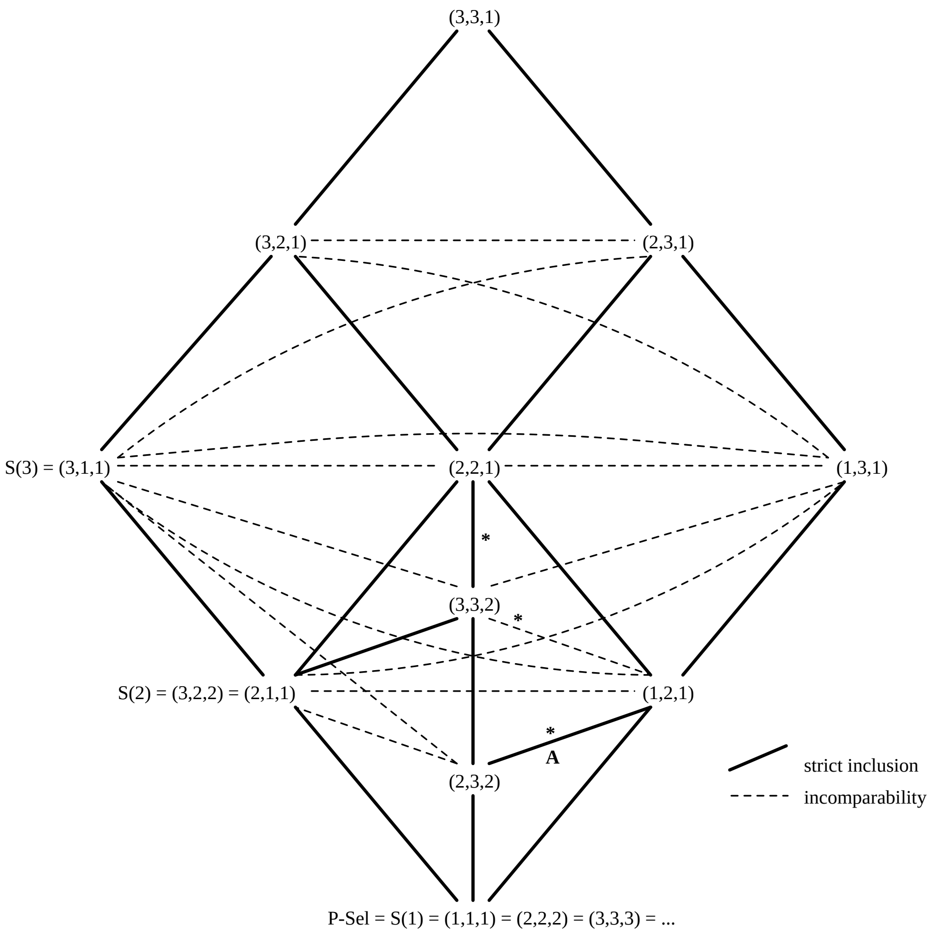

Figure 3: Relations between all nontrivial classes $\mathrm{GC}(b, c, d)$ with $1 \leq b, c, d \leq 3$.

References

[ABG90] A. Amir, R. Beigel, and W. Gasarch. Some connections between bounded query classes and non-uniform complexity. In *Proceedings of the 5th Structure in Complexity Theory Conference*, pages 232–243. IEEE Computer Society Press, July 1990. An extended manuscript, with the same authors and title, is dated July 11, 1994.

[BBS86] J. Balcázar, R. Book, and U. Schöning. Sparse sets, lowness and highness. *SIAM Journal on Computing*, 15(3):739–746, 1986.

[BKS95] R. Beigel, M. Kummer, and F. Stephan. Approximable sets. *Information and Computation*, 120(2):304–314, 1995.

[BT96] H. Buhrman and L. Torenvliet. P-selective self-reducible sets: A new characterization of P. *Journal of Computer and System Sciences*, 53(2):210–217, 1996.

[CH89] J. Cai and L. Hemachandra. Enumerative counting is hard. *Information and Computation*, 82(1):34–44, 1989.

[DHHT94] D. Denny-Brown, Y. Han, L. Hemaspaandra, and L. Torenvliet. Semi-membership algorithms: Some recent advances. *SIGACT News*, 25(3):12–23, September 1994.

[HHN+95] L. Hemaspaandra, A. Hoene, A. Naik, M. Ogiwara, A. Selman, T. Thierauf, and J. Wang. Nondeterministically selective sets. *International Journal of Foundations of Computer Science*, 6(4):403–416, 1995.

[HHSY91] L. Hemachandra, A. Hoene, D. Siefkes, and P. Young. On sets polynomially enumerable by iteration. *Theoretical Computer Science*, 80(2):203–226, 1991.

[HJ95] L. Hemaspaandra and Z. Jiang. P-selectivity: Intersections and indices. *Theoretical Computer Science*, 145(1–2):371–380, 1995.

[HJRW96] L. Hemaspaandra, Z. Jiang, J. Rothe, and O. Watanabe. The join can lower complexity. In *Proceedings of the 2nd Annual International Computing and Combinatorics Conference*, pages 260–267. Springer-Verlag *Lecture Notes in Computer Science #1090*, June 1996.

[HNOS96] L. Hemaspaandra, A. Naik, M. Ogihara, and A. Selman. Computing solutions uniquely collapses the polynomial hierarchy. *SIAM Journal on Computing*, 25(4):697–708, 1996.

[HU79] J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.

[Joc68] C. Jockusch. Semirecursive sets and positive reducibility. *Transactions of the AMS*, 131(2):420–436, 1968.

[KL80] R. Karp and R. Lipton. Some connections between nonuniform and uniform complexity classes. In *Proceedings of the 12th ACM Symposium on Theory of Computing*, pages 302–309, April 1980. An extended version has also appeared as: Turing machines that take advice, *L'Enseignement Mathématique*, 2nd series 28, 1982, pages 191–209.

[Ko83] K. Ko. On self-reducibility and weak P-selectivity. *Journal of Computer and System Sciences*, 26:209–221, 1983.

[Köb94] J. Köbler. Locating P/poly optimally in the extended low hierarchy. *Theoretical Computer Science*, 134:263–285, 1994.

[Köb95] J. Köbler. On the structure of low sets. In *Proceedings of the 10th Structure in Complexity Theory Conference*, pages 246–261. IEEE Computer Society Press, June 1995.

[KS85] K. Ko and U. Schöning. On circuit-size complexity and the low hierarchy in NP. *SIAM Journal on Computing*, 14(1):41–51, 1985.

[LS95] T. Long and M. Sheu. A refinement of the low and high hierarchies. *Mathematical Systems Theory*, 28:299–327, 1995.

[MS72] A. Meyer and L. Stockmeyer. The equivalence problem for regular expressions with squaring requires exponential space. In *Proceedings of the 13th IEEE Symposium on Switching and Automata Theory*, pages 125–129, 1972.

[Nic94] A. Nickelsen, 1994. Personal Communication.

[Ogi95] M. Ogihara. Polynomial-time membership comparable sets. *SIAM Journal on Computing*, 24(5):1068–1081, 1995.

[Rao94] R. Rao, 1994. Personal Communication.

[Rot95] J. Rothe. *On Some Promise Classes in Structural Complexity Theory*. PhD thesis, Institut für Informatik, Friedrich-Schiller-Universität Jena, Jena, Germany, 1995.

[Sch83] U. Schöning. A low and a high hierarchy within NP. *Journal of Computer and System Sciences*, 27:14–28, 1983.

[Sel79] A. Selman. P-selective sets, tally languages, and the behavior of polynomial time reducibilities on NP. *Mathematical Systems Theory*, 13:55–65, 1979.

[Sto77] L. Stockmeyer. The polynomial-time hierarchy. *Theoretical Computer Science*, 3:1–22, 1977.