Gap-Definability as a Closure Property

Stephen Fenner*
University of Southern Maine
Computer Science Department
96 Falmouth St., Portland, ME 04103

Lance Fortnow† Lide Li‡
University of Chicago
Department of Computer Science
1100 E. 58th St., Chicago, IL 60637

August 16, 1996

Abstract

Gap-definability and the gap-closure operator were defined in [FFK94]. Few complexity classes were known at that time to be gap-definable. In this paper, we give simple characterizations of both gap-definability and the gap-closure operator, and we show that many complexity classes are gap-definable, including $\mathbf{P}^{\#\mathbf{P}}$, $\mathbf{P}^{\#\mathbf{P}[1]}$, **PSPACE**, **EXP**, **NEXP**, **MP** (Middle-bit **P**), and $\mathbf{BP}\cdot\oplus\mathbf{P}$. If a class is closed under union, intersection and contains $\emptyset$ and Σ^*, then it is gap-definable if and only if it contains **SPP**; its gap-closure is the closure of this class together with **SPP** under union and intersection. On the other hand, we give some examples of classes which are reasonable gap-definable but not closed under union (resp. intersection, complement). Finally, we show that a complexity class such as **PSPACE** or **PP**, if it is not equal to **SPP**, contains a maximal gap-definable many-one reduction-closed subclass, which is properly between **SPP** and the class of all **PSPACE**-incomplete (**PP**-incomplete) sets with respect to containment. The gap-closure of the class of all incomplete sets in **PSPACE** (resp. **PP**) is **PSPACE** (resp. **PP**).

1 Introduction

In 1979, Valiant [Val79] defined the class $\#\mathbf{P}$, the class of functions definable as the number of accepting computations of some polynomial-time nondeterministic Turing machine. Valiant showed many natural problems complete for this class, including the permanent of a zero-one matrix. Toda [Tod91] showed that these functions have more power than previously believed; he showed how to reduce any problem in the polynomial-time hierarchy to a single value of a $\#\mathbf{P}$ function.

The class $\#\mathbf{P}$ has its shortcomings, however. In particular, $\#\mathbf{P}$ functions cannot take on negative values and thus $\#\mathbf{P}$ is not closed under subtraction. Also, one cannot express as a $\#\mathbf{P}$ function the permanent of a matrix with arbitrary (possibly negative) integer entries, or even a simple polynomial-time function which outputs negative values. Since much of the current work on counting classes is algebraic in nature, it is reasonable to seek an alternative to $\#\mathbf{P}$ with nicer algebraic properties.

Fenner, Fortnow and Kurtz [FFK94] analyzed the class **GapP**, a function class consisting of differences—"gaps"—between the number of accepting and rejecting paths of **NP** Turing machines. This class is exactly the closure of $\#\mathbf{P}$ under subtraction. **GapP** also has all the other nice closure properties of $\#\mathbf{P}$, such as

*Partially Supported by NSF Grant CCR-9209833. E-mail: fenner@usm.maine.edu.
†Partially Supported by NSF Grant CCR-9009936 and RCD-9253582. E-mail: fortnow@cs.uchicago.edu.
‡Partially Supported by NSF Grant RCD-9253582. Current address: Lehman Brothers Inc., 190 S. Lasalle Street, Suite 2500, Chicago, IL 60603. Email: lideli@lehman.com.

addition, multiplication, and binomial coefficients. Beigel, Reingold and Spielman first used gaps to great advantage in [BRS95] to show that **PP** is closed under intersection. Toda and Ogiwara have also formulated their results in [TO92] using **GapP** instead of **#P**.

Fenner, Fortnow and Kurtz looked at classes such as **PP**, $\mathbf{C_{=}P}$, $\oplus$**P** and **SPP** that can be defined in terms of **GapP** functions. They defined a natural general notion of gap-definability and also defined GapCl, a non-constructive closure operation on sets (the 'gap-closure'). They showed that any countable set of languages $\mathcal{C}$ has a unique minimum gap-definable class GapCl($\mathcal{C}$) containing it. However, their definition does not yield an easy way to determine properties of gap definable classes or to determine which classes may be gap-definable.

In this paper we will take the mystery out of gap-definability. In Section 3 we give a simple characterization of gap-closure. By the definition of gap-definability, there is no restriction on the accepting set A and the rejecting set R. We show that A and R can be chosen to be recursive under reasonable circumstances. We use the results of the previous sections to describe some properties of the gap-closure and gap-definability. We show that Boolean closure properties such as closure under union or intersection are not necessary for gap-definability. In Section 4 we give a simple characterization of when classes are gap-definable. Using this characterization we show many common classes, such as $\mathbf{P^{\#P}}$, $\mathbf{P^{\#P[1]}}$, **PSPACE**, **EXP**, **NEXP**, **MP** and **BP**$\cdot\oplus$**P**, are gap-definable. In general, for complexity classes with some reasonable restrictions, we give simple necessary and sufficient conditions for whether they are gap-definable. In Section 5, we show that a complexity class such as **PSPACE** or **PP**, if it is not equal to **SPP**, contains a maximal gap-definable many-one reduction-closed subclass, which is properly between **SPP** and the class of all **PSPACE**-incomplete (**PP**-incomplete) sets with respect to containment. The gap-closure of the class of all incomplete sets in **PSPACE** (resp. **PP**) is **PSPACE** (resp. **PP**).

2 Preliminaries

Definition 2.1 *A* counting machine (CM) *is a nondeterministic Turing machine running in polynomial time with two halting states: accepting and rejecting, and every computation path must end in one of these states. An oracle machine having the above properties and running in polynomial time uniformly for all oracles is called an* oracle counting machine (OCM).

If M is a counting machine, then $acc_M(x)$ (resp. $rej_M(x)$) are the number of accepting (resp. rejecting) paths of M on input x. We also define

$$gap_M = acc_M - rej_M.$$

Two different types of *counting functions* were defined via counting machines.

Definition 2.2 *[Val79]*

$$\#\mathbf{P} = \{acc_M \mid M \textit{ is a CM}\}$$

$$\mathbf{UP} = \{L \mid \chi_L \in \#\mathbf{P}\}$$

Definition 2.3 *[FFK94]*

$$\mathbf{GapP} = \{gap_M \mid M \textit{ is a CM}\}$$

$$\mathbf{SPP} = \{L \mid \chi_L \in \mathbf{GapP}\}$$

Many complexity classes were originally defined via CMs, together with criteria on the total number of computational paths and accepting paths. Fenner, Fortnow and Kurtz observed that some of them such as **PP**, $\mathbf{C_{=}P}$, $\mathbf{Mod_{\mathit{k}}P}$ can be characterized via **GapP** functions. They initiated a concept called *gap-definability.*

Definition 2.4 *[FFK94] A class $\mathcal{C}$ of languages is* gap-definable *if there exist disjoint sets $A, R \subseteq \Sigma^* \times \mathbf{Z}$ such that, for any language L, $L \in \mathcal{C}$ if and only if there exists a CM M with*

$$\begin{aligned} x \in L &\implies (x, \mathrm{gap}_M(x)) \in A \\ x \notin L &\implies (x, \mathrm{gap}_M(x)) \in R, \end{aligned}$$

for all $x \in \Sigma^*$. Since A and R uniquely determine $\mathcal{C}$, we let $\mathbf{Gap}(A,R)$ *denote the class $\mathcal{C}$. A function $g \in$* $\mathbf{GapP}$ *is called* (A,R)-proper *if $\forall x, (x, g(x)) \in A \cup R$. A gap-definable class $\mathcal{C}$ is called* reasonable *if $\mathcal{C}$ contains $\emptyset$ and Σ^*.*

Not all gap-definable classes are reasonable; for example any class consisting of a single language is a nonreasonable gap-definable class.

In [FFK94], it has been shown that if $\mathcal{C}$ is gap-definable, then $\mathcal{C}$ is reasonable if and only if $\mathbf{SPP} \subseteq \mathcal{C}$. Since in most cases we are only interested in reasonable classes, the above statement means that **SPP** is the minimal gap-definable class.

Some complexity classes, such as those we listed above, even deserve simpler characterization. We call a class *simply* gap-definable if it is gap-definable and A, R in Definition 2.4 depend on $\mathrm{gap}_M(x)$ only. Using a proposition in [FFK94], the classes **PP**, $\mathbf{C_{=}P}$, and $\mathbf{Mod}_k\mathbf{P}$ (for $k \geq 2$) can be redefined as the following:

1. ([Gil77], [Sim75])
   $L \in \mathbf{PP} \iff (\exists g \in \mathbf{GapP})(\forall x)[x \in L \leftrightarrow g(x) > 0]$.

2. ([Sim75], [Wag86])
   $L \in \mathbf{C_{=}P} \iff (\exists g \in \mathbf{GapP})(\forall x)[x \in L \leftrightarrow g(x) = 0]$.

3. ([CH90][Her90][BG92])
   $L \in \mathbf{Mod}_k\mathbf{P} \iff (\exists g \in \mathbf{GapP})(\forall x)[x \in L \leftrightarrow g(x) \not\equiv 0 \bmod k]$.

Every gap-definable class is countable, but the converse does not hold. However, Fenner, Fortnow and Kurtz [FFK94] showed that every countable class is contained in a unique minimum gap-definable class, called its *gap-closure*. They showed that the gap-closure of a countable class $\mathcal{D}$, denoted as $\mathrm{GapCl}(\mathcal{D})$, can be constructed as the following:

Definition 2.5 *[FFK94] Let the class $\mathcal{D} = \{L_1, L_2, L_3, \ldots\}$ be a countable collection of languages and $W = \{w_1, w_2, \ldots\}$ be an immune set, i.e., W is infinite with no infinite recursively enumerable subset. Define*

$$A_{\mathcal{D}} = \{(x, w_i) \mid x \in L_i\}$$

and

$$R_{\mathcal{D}} = \{(x, w_i) \mid x \notin L_i\},$$

and define $\mathrm{GapCl}(\mathcal{D}) = \mathbf{Gap}(A_{\mathcal{D}}, R_{\mathcal{D}})$.

In Theorem 3.3 and Corollary 3.4 we show that immune sets are not required to represent $\mathrm{GapCl}(\mathcal{C})$ as a gap-definable class.

Certain "delta" functions were defined in [FFK94]. For integers k and B with $0 \leq k \leq B$, define

$$\delta_k^B(x) = \binom{x}{k}\binom{B-x}{B-k}$$

For integers k and B with $0 \leq k \leq B$, we have

$$\delta_k^B(x) = \begin{cases} 0 & \text{if } 0 \leq x < k, \\ 1 & \text{if } x = k, \\ 0 & \text{if } k < x \leq B. \end{cases}$$

If $f \in \mathbf{GapP}$, then $\delta_k^B \circ f \in \mathbf{GapP}$ [FFK94].

Using the delta functions, we can show the following lemma

Lemma 2.6 *If $f \in$* $\mathbf{GapP}$ *and $range(f) \in \{w_1, \ldots, w_k\}$, then the set $S_i = \{x \mid f(x) = w_i\} \in$* $\mathbf{SPP}$ *for $i = 1, \ldots, k$.*

Proof: Let $B = \max\{w_1, \ldots, w_k\}$ and let $f_i = \delta^B_{w_i} \circ f$ for $1 \le i \le k$. Each f_i is in **GapP**. For all $x \in \Sigma^*$, we have

$$f_i(x) = \delta^B_{w_i}(f(x)) = \begin{cases} 1 & \text{if } f(x) = w_i, \\ 0 & \text{otherwise.} \end{cases}$$

Thus f_i witnesses that $S_i \in \mathbf{SPP}$. □

Some operators based on counting are often used to define certain complexity classes. Here we list some of them.

Definition 2.7 *1. $L \in \exists \cdot B$ for some set B iff there is a polynomial p such that $x \in L$ iff $\exists y \in \{0,1\}^{p(|x|)}, x\#y \in B$. For class $\mathcal{C}$, we define $\exists \cdot \mathcal{C} = \bigcup_{B \in \mathcal{C}} \exists \cdot B$.*

2. $L \in \mathbf{BP}\cdot\mathcal{C}$ iff there is $C \in \mathcal{C}$ and a polynomial p s.t.

$$\begin{aligned} x \in L &\implies Pr\{y \in \{0,1\}^{p(|x|)} \mid x\#y \in C\} > 2/3 \\ x \notin L &\implies Pr\{y \in \{0,1\}^{p(|x|)} \mid x\#y \in C\} < 1/3 \end{aligned}$$

The class $\mathbf{C}\cdot\mathcal{C}$ is defined in the same way except we use 1/2 instead of 2/3 and 1/3.

3. $L \in \mathbf{R}\cdot\mathcal{C}$ iff there is $C \in \mathcal{C}$ and a polynomial p s.t.

$$\begin{aligned} x \in L &\implies Pr\{y \in \{0,1\}^{p(|x|)} \mid x\#y \in C\} \ge 1/2 \\ x \notin L &\implies Pr\{y \in \{0,1\}^{p(|x|)} \mid x\#y \in C\} = 0 \end{aligned}$$

In this paper, we use "$\subseteq$" to mean "is contained in" and "$\subset$" to mean "is properly contained in".

3 Gap-Closures and Boolean Closures

The following theorem provides a simplified characterization of the gap-closure operator, GapCl.

Theorem 3.1 *Let $\mathcal{C}$ be a countable class of languages and L an arbitrary language. $L \in \mathrm{GapCl}(\mathcal{C})$ if and only if there exist $L_1, \ldots, L_k \in \mathcal{C}$ and $S_1, \ldots, S_k \in \mathbf{SPP}$ such that*

1. *$S_i \cap S_j = \emptyset$ for all $i \neq j$,*
2. *$\bigcup_{i=1}^k S_i = \Sigma^*$, and*
3. *$L = \bigcup_{i=1}^k (L_i \cap S_i)$.*

Proof: Fix an immune set $W = \{w_1, w_2, w_3, \ldots\}$ and an enumeration $L_1, L_2, L_3, \ldots$ of the languages in $\mathcal{C}$. As in Definition 2.5, we define $A \stackrel{\text{df}}{=} \{(x, w_i) \mid x \in L_i\}$, and $R \stackrel{\text{df}}{=} \{(x, w_i) \mid x \notin L_i\}$. By definition, $\mathrm{GapCl}(\mathcal{C}) = \mathbf{Gap}(A, R)$. First, suppose $L \in \mathbf{Gap}(A, R)$. Then there is an $f \in \mathbf{GapP}$ such that $\mathrm{range}(f) \subseteq W$ and for all $x \in \Sigma^*$, $x \in L \iff (x, f(x)) \in A$. Note that $(x, f(x)) \in R$ if $x \notin L$. Since W is immune, there is a k such that $\mathrm{range}(f) \subseteq \{w_1, \ldots, w_k\}$. For $1 \le i \le k$ we define $S_i \stackrel{\text{df}}{=} \{x \mid f(x) = w_i\}$. By Lemma 2.6, $S_i \in \mathbf{SPP}$ for $i = 1, 2, \ldots, k$. The sets $S_1, \ldots, S_k$ satisfy the first two conditions of the theorem. To show that the third condition is satisfied, we note that for all x, $x \in L \iff (x, f(x)) \in A \iff (\exists i,\ 1 \le i \le k)[f(x) = w_i \ \&\ x \in L_i] \iff x \in \bigcup_{i=1}^k (L_i \cap S_i)$.

Conversely, suppose there exist $S_1, \ldots, S_k \in \mathbf{SPP}$ such that L satisfies the three conditions of the theorem. For $1 \le i \le k$, let $f_i(x)$ be the characteristic function of S_i, and define $f(x) \stackrel{\text{df}}{=} \sum_{i=1}^k f_i(x) \cdot w_i$. The function $f_i \in \mathbf{GapP}$ and $f \in \mathbf{GapP}$. By the first two conditions, we see that for any given x, $f(x) = w_{i(x)}$ where $i(x)$ is the unique i such that $x \in S_i$. Thus the graph of f is contained in $A \cup R$. By the third condition, we have: $x \in L \iff (\exists i,\ 1 \le i \le k)[x \in L_i \cap S_i] \iff x \in L_{i(x)} \cap S_{i(x)} \iff (x, w_{i(x)}) \in A \iff (x, f(x)) \in A$. Therefore, f witnesses that $L \in \mathbf{Gap}(A, R)$. This completes the proof. □

In the definition of gap-definability (Definition 2.4), there is no restriction on the accepting set A and rejecting set R. In fact, in [FFK94] it seemed that we needed $A \cup R$ to be nonrecursive—defined using an immune set—to prove properties of the GapCl operator. We will now show that A and R can always be chosen such that $A \cup R$ is recursive, and under reasonable circumstances, both A and R themselves are recursive.

We say that a function f is *covered* by g if $(\forall x)(\exists i)[f(x) = g(x,i)]$.

Lemma 3.2 *There is a recursive function* $g : \Sigma^* \times \mathbf{N} \rightarrow \mathbf{N}$ *such that*

1. $\forall x, i, \ g(x,i) \in \{2i, 2i+1\}$.
2. *For any fixed* i, $g(x,i) = 2i$ *for all but finitely many* x.
3. *If* $f \in \mathbf{GapP}$ *is covered by* g, *then* $range(f)$ *is finite.*

Proof: Let $h(i,x) = f_i(x)$ be a universal function for **GapP**, i.e., $\mathbf{GapP} = \{f_0, f_1, f_2, \ldots\}$. There is a canonical linear ordering on $\Sigma^* \times \mathbf{N}$. We define g in stages: Initially all the f_i are unmarked. At stage (x,i), if there exists $j \leq i$ such that f_j is unmarked and $f_j(x) \in \{2i, 2i+1\}$, choose the smallest such j, mark function f_j, and set $g(x,i) = 4i + 1 - f_j(x)$; otherwise set $g(x,i) = 2i$. The function g is recursive, and for all $x, g(x,i) \in \{2i, 2i+1\}$. Since $g(x,i) = 2i+1$ only if there is a $j \leq i$ such that f_j is marked at stage (x,i), and since each f_j is marked at most once, $g(x,i) = 2i$ for all but finitely many x. It remains to show part 3.

Suppose $f = f_i$ has no upper-bound. There is a stage s after which no more **GapP** functions prior to f will be marked. Since f is unbounded, we can find x such that $f(x) \in \{2j, 2j+1\}$ with $(x,j) > s$ and $j \geq i$. Then f must be marked at stage (x,j) if it has not already been marked. Suppose f is marked at stage (y,k), then $f(y) \neq g(y,k)$ by the definition of g. Note also that $f(y) \in \{2k, 2k+1\}$ but $g(y,k') \notin \{2k, 2k+1\}$ for any $k' \neq k$. Thus f is not covered by g. In other words, if f is covered by g, then f is upper-bounded, and therefore $range(f)$ is finite since we also have $f(x) \geq 0$ for all x. $\square$

Theorem 3.3 *For any countable class* $\mathcal{C}$, *there are sets* A, R *such that* $A \cup R$ *is recursive and* $\mathrm{GapCl}(\mathcal{C}) = \mathbf{Gap}(A,R)$.

Proof: Let $\mathcal{C} = \{L_1, L_2, \ldots\}$ and g be the function defined in the lemma. Set $A = \{(x, g(x,i)) | x \in L_i\}$ and $R = \{(x, g(x,i)) | x \notin L_i\}$. Given (x,m), $(x,m) \in A \cup R$ iff there is i such that $g(x,i) = m$. Since $g(x,i) \in \{2i, 2i+1\}$ and g is recursive, we have that $A \cup R$ is recursive.

Now we show that $\mathcal{C} \subseteq \mathbf{Gap}(A,R)$. Let $L_i \in \mathcal{C}$. Define a function h such that $\forall x, h(x) = g(x,i)$. Since $g(x,i) = 2i$ for all but finitely many x, we have $h(x) \in \mathbf{FP} \subseteq \mathbf{GapP}$. We then have $x \in L_i \Longrightarrow (x, h(x)) = (x, g(x,i)) \in A$ and $x \notin L_i \Longrightarrow (x, h(x)) = (x, g(x,i)) \in R$. So $L_i \in \mathbf{Gap}(A,R)$. Thus $\mathcal{C} \subseteq \mathbf{Gap}(A,R)$. Since $\mathbf{Gap}(A,R)$ is gap-definable, we have $\mathrm{GapCl}(\mathcal{C}) \subseteq \mathbf{Gap}(A,R)$.

Conversely, let $L \in \mathbf{Gap}(A,R)$. That is, there is $f \in \mathbf{GapP}$ such that

$x \in L \Longrightarrow (x, f(x)) \in A \Longrightarrow f(x) = g(x,i)$ for some i and $x \in L_i$

$x \notin L \Longrightarrow (x, f(x)) \in R \Longrightarrow f(x) = g(x,i)$ for some i and $x \notin L_i$

The function f is covered by g, so $range(f)$ is finite by Lemma 3.2. Let $range(f) = \{n_1, n_2, \ldots, n_r\}$. By arguments similar to those in Theorem 3.1, it is not hard to see that $S_i \stackrel{\mathrm{df}}{=} \{x | f(x) = n_i\} \in \mathbf{SPP}$, and $L = \bigcup_{i=1}^{r} (S_i \cap L_{\lfloor n_i/2 \rfloor})$. Also note that $\{S_1, S_2, \ldots, S_r\}$ is a partition of Σ^*, so we have $L \in \mathrm{GapCl}(\mathcal{C})$ by Theorem 3.1. This completes the proof. $\square$

Corollary 3.4 *If there is a universal recursive enumeration of* $\mathcal{C}$, *then there are recursive sets* A *and* R *such that* $\mathrm{GapCl}(\mathcal{C}) = \mathbf{Gap}(A,R)$. *(For example,* $\mathrm{GapCl}(\mathbf{NP}) = \mathbf{Gap}(A,R)$ *for some recursive sets* A *and* R.*)* $\square$

It was shown in [FFK94] that $\mathrm{GapCl}(\mathcal{C})$ inherits many closure properties of $\mathcal{C}$. Here we add to that list, and obtain as a corollary a simple characterization of $\mathrm{GapCl}(\mathcal{C})$ for many common classes $\mathcal{C}$.

Lemma 3.5 1. *If class* $\mathcal{C}$ *is closed under union, then so is* $\mathrm{GapCl}(\mathcal{C})$.

2. *If class* $\mathcal{C}$ *is closed under intersection, then so is* $\mathrm{GapCl}(\mathcal{C})$.

Proof:

1. Let $\mathcal{C} = \{L_1, L_2, \ldots\}$ be closed under union and $L_a, L_b \in \mathrm{GapCl}(\mathcal{C})$. $\exists S_1, S_2, \ldots, S_m, T_1, T_2, \ldots, T_n \in$ **SPP** where $S_i \cap S_j = T_i \cap T_j = \emptyset$ for $i \neq j$ and $\bigcup_{i=1}^{m} S_i = \bigcup_{i=1}^{n} T_i = \Sigma^*$ s.t. $L_a = \bigcup_{i=1}^{m} (L_i \cap S_i), L_b =$

$\bigcup_{j=1}^n (L_j \cap T_j)$, (Theorem 3.1). For $i, j, 1 \le i \le m, 1 \le j \le n$, let $Q_{ij} = S_i \cap T_j$. Note $Q_{ij} \in \mathbf{SPP}$ since **SPP** languages are closed under intersection [FFK94]. Then $S_i = \bigcup_{j=1}^n Q_{ij}, T_j = \bigcup_{i=1}^m Q_{ij}$.

$$\begin{aligned} L_a &= \bigcup_{i=1}^m [L_i \cap (\bigcup_{j=1}^n Q_{ij})] = \bigcup_{i=1}^m \bigcup_{j=1}^n (L_i \cap Q_{ij}) \\ L_b &= \bigcup_{j=1}^n [L_j \cap (\bigcup_{i=1}^m Q_{ij})] = \bigcup_{j=1}^n \bigcup_{i=1}^m (L_j \cap Q_{ij}) \end{aligned}$$

Since $(L_i \cap Q_{ij}) \cup (L_j \cap Q_{ij}) = (L_i \cup L_j) \cap Q_{ij}, L_a \cup L_b = \bigcup_{i=1}^m \bigcup_{j=1}^n [(L_i \cup L_j) \cap Q_{ij}]$. Note that $L_i \cup L_j \in \mathcal{C}$, and $\{Q_{ij}\}_{1\le i\le m, 1\le j\le n}$ is a partition of Σ^*, thus we have $L_a \cup L_b \in \mathrm{GapCl}(\mathcal{C})$.

2. Similarly, since $(L_i \cap Q_{ij}) \cap (L_j \cap Q_{ij}) = (L_i \cap L_j) \cap Q_{ij}$, and $(L_i \cap Q_{ij}) \cap (L_l \cap Q_{kl}) = \emptyset$ for $(i,j) \neq (k,l)$, applying the distributive law, we have $L_a \cap L_b = \bigcup_{i=1}^m \bigcup_{j=1}^n [(L_i \cap L_j) \cap Q_{ij}]$. Again, since $L_i \cap L_j \in \mathcal{C}$ and $\{Q_{ij}\}_{1\le i\le m, 1\le j\le n}$ is a partition of Σ^*, we have $L_a \cap L_b \in \mathrm{GapCl}(\mathcal{C})$.

□

For some classes which seem unlikely to be gap-definable, such as **NP** and **BPP**, we want to know what their gap-closures are. The following corollary gives us a simple way to describe them. For example, GapCl(**NP**) is exactly the closure of $\mathbf{NP} \cup \mathbf{SPP}$ under union and intersection.

Corollary 3.6 *If $\mathcal{C}$ is closed under union and intersection, and $\{\emptyset, \Sigma^*\} \subseteq \mathcal{C}$, then* $\mathrm{GapCl}(\mathcal{C})$ *is the closure under union and intersection of $\mathcal{C} \cup \mathbf{SPP}$.*

Proof: Since $\{\emptyset, \Sigma^*\} \subseteq \mathcal{C} \subseteq \mathrm{GapCl}(\mathcal{C}), \mathbf{SPP} \subseteq \mathrm{GapCl}(\mathcal{C})$. Let $\mathcal{D}$ be the closure of $\mathcal{C} \cup \mathbf{SPP}$ under union and intersection. By Lemma 3.5, $\mathrm{GapCl}(\mathcal{C})$ is closed under union and intersection, and $\mathcal{C} \cup \mathbf{SPP} \subseteq \mathrm{GapCl}(\mathcal{C})$, so we have $\mathcal{D} \subseteq \mathrm{GapCl}(\mathcal{C})$. Conversely, $\mathrm{GapCl}(\mathcal{C}) \subseteq \mathcal{D}$ by Theorem 3.1. □

4 Gap-definability

After gap-definability was defined, **PP**, $\mathbf{C_=P}$ and $\mathbf{Mod}_k\mathbf{P}$ were then the only well-studied classes shown to be gap-definable. Many gap-definable classes do not seem to be directly related to **GapP**, while those that can be easily settled, such as the above three, are indeed simply gap-definable.

An interesting fact is that gap-definability and certain Boolean closure properties are closely related. In this section, we provide a simple characterization of gap-definability. It requires only Boolean operations, without involving any **GapP** functions. Then we show that many commonly discussed classes are indeed gap-definable.

We now prove a theorem that characterizes gap-definability by a certain Boolean closure property with **SPP**. It will play an important role in this section.

Theorem 4.1 *Let $\mathcal{C}$ be any countable class of languages. The following are equivalent:*

1. *$\mathcal{C}$ is gap-definable;*
2. *For all $L_1, L_2 \in \mathcal{C}$ and $S \in \mathbf{SPP}$, $(L_1 \cap S) \cup (L_2 \cap \overline{S}) \in \mathcal{C}$.*

Proof: $(1 \Longrightarrow 2)$: Let $\mathcal{C}$ be gap-definable, $L_1, L_2 \in \mathcal{C}$ and $S \in \mathbf{SPP}$. There exist disjoint $A, R \subseteq \Sigma^* \times \mathbf{Z}$ and $g_1, g_2 \in \mathbf{GapP}$ such that for $i \in \{1, 2\}$,

$$\begin{aligned} x \in L_i &\Longrightarrow (x, g_i(x)) \in A \\ x \notin L_i &\Longrightarrow (x, g_i(x)) \in R \end{aligned}$$

and there exist $f \in \mathbf{GapP}$ such that

$$\begin{aligned} x \in S &\Longrightarrow f(x) = 1 \\ x \notin S &\Longrightarrow f(x) = 0. \end{aligned}$$

We define the **GapP** function

$$h = g_1 f + g_2(1 - f).$$

Fix an input x. We have four cases.

1. $x \in S \cap L_1$: $(x, h(x)) = (x, g_1(x)) \in A$.
2. $x \in S - L_1$: $(x, h(x)) = (x, g_1(x)) \in R$.
3. $x \in \overline{S} \cap L_2$: $(x, h(x)) = (x, g_2(x)) \in A$.
4. $x \in \overline{S} - L_2$: $(x, h(x)) = (x, g_2(x)) \in R$.

This shows that $(L_1 \cap S) \cup (L_2 \cap \overline{S}) \in \mathcal{C}$, as witnessed by h.

($2 \Longrightarrow 1$): We show that $\mathrm{GapCl}(\mathcal{C}) = \mathcal{C}$. Let $\mathcal{C} = \{L_1, L_2, \ldots\}, L \in \mathrm{GapCl}(\mathcal{C})$. Theorem 3.1 shows that there exist pairwise disjoint $S_1, S_2, \ldots, S_r \in$ **SPP** such that $\bigcup_{i=1}^r S_i = \Sigma^*$ and $L = \bigcup_{i=1}^r (S_i \cap L_i)$. Now we show that $L \in \mathcal{C}$ by induction on r. If $r = 1$ then $S_1 = \Sigma^*$, so $L = L_1 \in \mathcal{C}$.

Suppose $r > 1$. Let $L' = \bigcup_{i=2}^r (S_i \cap L_i)$. Let $S_2' = S_1 \cup S_2$ and $S_i' = S_i$ for $i > 2$. Let $L'' = \bigcup_{i=2}^r (S_i' \cap L_i)$. We have

$$\begin{aligned} L &= (S_1 \cap L_1) \cup (\bigcup_{i=2}^r (S_i \cap L_i)) \\ &= (S_1 \cap L_1) \cup L' \\ &= (S_1 \cap L_1) \cup (\overline{S_1} \cap L'') \end{aligned}$$

By the induction hypothesis, $L'' \in \mathcal{C}$. By condition 2, we have $L \in \mathcal{C}$. □

Many interesting complexity classes are closed under union and intersection, and contain $\{\emptyset, \Sigma^*\}$. For them, the question of gap-definability is just a matter of whether or not they contain **SPP**.

Corollary 4.2 *If $\mathcal{C}$ is closed under union and intersection, and $\{\emptyset, \Sigma^*\} \subseteq \mathcal{C}$, then $\mathcal{C}$ is gap-definable if and only if* **SPP** $\subseteq \mathcal{C}$.

Proof: If $\mathcal{C}$ is gap-definable and $\{\emptyset, \Sigma^*\} \subseteq \mathcal{C}$, then **SPP** $\subseteq \mathcal{C}$ [FFK94]. Conversely, if $\mathcal{C}$ is closed under union and intersection, and **SPP** $\subseteq \mathcal{C}$, then we have condition 2 in Theorem 4.1. So $\mathcal{C}$ is gap-definable. □

Corollary 4.3 *If $\mathcal{C} \subseteq \mathcal{D}$, $\mathcal{D}$ is closed under union and intersection, and $\mathcal{C}$ contains $\{\emptyset, \Sigma^*\}$, then $\mathcal{C}$ is gap-definable implies that $\mathcal{D}$ is gap-definable.*

This corollary says that for many classes, each would be gap-definable if one of its subclasses is gap-definable. For example, **BPP**, or $\boldsymbol{\Sigma}_k$ for any k is not gap-definable unless **PH** is gap-definable.

Definition 4.4 *Let A be an oracle set or an oracle function. $\mathbf{P}^A$ denotes the class of languages accepted by a deterministic polynomial-time oracle machine with oracle A. Let $\mathbf{P}^{\mathcal{C}} = \bigcup_{A \in \mathcal{C}} \mathbf{P}^A$. $\mathbf{P}^{\mathcal{C}[1]}$ is defined in the same way but the machine is allowed to ask only one oracle question in each computation. The class $\mathbf{NP}^{\mathcal{C}}$ is defined similarly.*

Corollary 4.5 *If $\mathcal{C}$ is a gap-definable class containing $\{\emptyset, \Sigma^*\}$, and closed under join, then $\mathbf{P}^{\mathcal{C}}$ and $\mathbf{NP}^{\mathcal{C}}$ are also gap-definable.*

Proof: If $\mathcal{C}$ is closed under join, then $\mathbf{P}^{\mathcal{C}}$ and $\mathbf{NP}^{\mathcal{C}}$ are closed under union and intersection. By the previous corollary, we get the conclusion. □

We are now able to prove that a number of well-known complexity classes are gap-definable.

Corollary 4.6

1. $\mathbf{P}^{\#\mathbf{P}}, \mathbf{PSPACE}, \mathbf{EXP}, \mathbf{NEXP}$ *are gap-definable;*
2. $\mathbf{P}^{\#\mathbf{P}[1]}$ *is gap-definable;*
3. **MP** *(Middle-bit* **P***) is gap-definable (For the definition of* **MP***, see [GKR+95]).*

Proof:

1. All these classes are closed under union and intersection, and all contain **SPP**. So they are gap-definable by Corollary 4.2

2. Note that $\mathbf{SPP} \subseteq \mathbf{P}^{\mathbf{GapP}[1]} \subseteq \mathbf{P}^{\#\mathbf{P}[1]}$ ([FFK94]). Cai and Hemachandra [CH89] show that a polynomial number of nonadaptive queries to $\#\mathbf{P}$ is equivalent to a single query of $\#\mathbf{P}$ and thus $\mathbf{P}^{\#\mathbf{P}[1]}$ is closed under union and intersection.

3. **MP** (Middle-bit **P**) was originally defined via $\#\mathbf{P}$ functions [GKR$^+$95]. One alternative definition for **MP**: $L \in \mathbf{MP}$ if and only if $\exists g \in \mathbf{GapP}$, and $h \in \mathbf{FP}$ s.t. $x \in L$ iff the $h(x)$-th bit in the binary representation of $g(x)$, which is called *midbit*, is 1. Here, the i-th bit of $g(x)$ is equal to

$$\left\lfloor \frac{g(x)}{2^i} \right\rfloor \bmod 2.$$

Note that this definition is meaningful for any integer value of $g(x)$. Let $L_1, L_2 \in \mathbf{MP}$ witnessed by g_1, g_2 and h_1, h_2 respectively. Let S be in **SPP** witnessed by a **GapP** function f. Define function $g = 2^{h_2} g_1 f + 2^{h_1} g_2 (1-f)$. Now we have that $x \in (L_1 \cap S) \cup (L_2 \cap \overline{S})$ if and only if the $h_1(x) + h_2(x)$-th bit of $g(x)$ is one.

This proves that $(L_1 \cap S) \cup (L_2 \cap \overline{S})$ is in **MP**. Thus **MP** is gap-definable by Theorem 4.1.

□

Recall that a class $\mathcal{C}$ of languages is *simply gap-definable* if there exist disjoint sets $A, R \subseteq \mathbf{Z}$ such that, for any language L, $L \in \mathcal{C}$ if and only if there exists a $g \in \mathbf{GapP}$ with

$$\begin{aligned} x \in L &\implies g(x) \in A \\ x \notin L &\implies g(x) \in R, \end{aligned}$$

The classes $\mathbf{PP}, \mathbf{C_=P}, \oplus\mathbf{P}$ and **SPP** are all simply gap-definable.

Theorem 4.7 *If $\mathcal{C}$ is simply gap-definable, then*

1. $\exists \cdot \mathcal{C}$ *is gap-definable.*

2. $\mathbf{BP}\cdot\mathcal{C}$, $\mathbf{C}\cdot\mathcal{C}$ *and* $\mathbf{R}\cdot\mathcal{C}$ *are gap-definable.*

Proof: 1) Let $\mathcal{C} = \{C_1, C_2, \ldots\}$. Since $\mathcal{C}$ is simply gap-definable, by definition, $\exists A, R \subseteq \mathbf{Z}, g_1, g_2, \ldots \in$ **GapP** such that

$$\begin{aligned} w \in C_i &\implies g_i(w) \in A \\ w \notin C_i &\implies g_i(w) \in R \end{aligned}$$

We assume R is not empty, otherwise the proof would be trivial. Let $\{L_{11}, L_{12}, \ldots\}$ be an enumeration of $\exists \cdot C_1$, $\{L_{21}, L_{22}, \ldots\}$ be an enumeration of $\exists \cdot C_2$, etc. The class $\exists \cdot \mathcal{C} = \cup_i \ \exists \cdot C_i$ can be listed in a certain way (e.g., use a pairing function), so that $\exists \cdot \mathcal{C} = \{L_1, L_2, \ldots\}$, and there exists a function ϕ in **FP** such that $L_i \in \ \exists \cdot C_{\phi(i)}$.

Let $L \in \mathrm{GapCl}(\exists \cdot \mathcal{C})$, and $W = \{w_1, w_2, \ldots\}$ be an immune set. By Definition 2.4 and Definition 2.5, there is a function $h \in \mathbf{GapP}$ s.t.

$$\begin{aligned} x \in L &\implies h(x) = w_t \ \&\ x \in L_t \text{ for some } t \\ x \notin L &\implies h(x) = w_t \ \&\ x \notin L_t \text{ for some } t \end{aligned}$$

and $range(h) = \{w_1, w_2, \ldots, w_r\}$ for some r such that $w_1 < w_2 < \ldots$ [FFK94]. Now we have polynomials $p_1, p_2, \ldots, p_r$ such that $x \in L_i$ iff $\exists y \in \{0,1\}^{p_i(|x|)}$ with $x\#y \in C_{\phi(i)}$ for $i = 1, 2, \ldots, r$. Choose a polynomial p such that $p(n) \geq p_i(n)$ for all $n \in \mathbf{Z}$ and $i = 1, 2, \ldots, r$.

Fix k in R. We define $\overline{g}_i$ as follows:

$$\overline{g}_i(x\#y) = \begin{cases} g_{\phi(i)}(x\#y_1) & \text{if } y = y_1y_2,\ |y_1| = p_i(|x|) \text{ and } |y| = p(|x|), \\ k & \text{otherwise} \end{cases}$$

We define a function f,

$$f(x\#y) = \sum_{i=1}^{r} \delta_{w_i}^{w_r}(h(x)) \cdot \overline{g}_i(x\#y)$$

and a set

$$B = \{x\#y \mid f(x\#y) \in A\}$$

The function $\overline{g}_i \in \mathbf{GapP}$ and is (A,R)-proper for $i = 1,2,\ldots,r$. One may verify that $f \in \mathbf{GapP}$. We need to show that f is an (A,R)-proper function, i.e., $f(w) \in A \cup R$ for all w. In fact

$$\begin{array}{rcl} x\#y \in B & \Longrightarrow & f(x\#y) \in A \\ x\#y \notin B & \Longrightarrow & f(x\#y) \notin A \ [Let\ h(x) = w_i, then\ f(x\#y) = \overline{g}_i(x\#y)] \\ & \Longrightarrow & \overline{g}_i(x\#y) \notin A \\ & \Longrightarrow & \overline{g}_i(x\#y) \in R \\ & \Longrightarrow & f(x\#y) \in R \end{array}$$

This also shows that $B \in \mathbf{Gap}(A,R) = \mathcal{C}$. Now,

$$\begin{array}{rcl} x \in L & \Longrightarrow & h(x) = w_t \text{ and } x \in L_t \text{ for some } t, (L_t \in \ \exists \cdot C_{\phi(t)}) \\ & \Longrightarrow & \exists y_1 \in \{0,1\}^{p_t(|x|)}\ s.t.\ x\#y_1 \in C_{\phi(t)} \\ & \Longrightarrow & \exists y_1 \in \{0,1\}^{p_t(|x|)}\ s.t.\ g_{\phi(t)}(x\#y_1) \in A \\ & \Longrightarrow & \exists y \in \{0,1\}^{p(|x|)}\ s.t.\ \overline{g}_t(x\#y) \in A \\ & \Longrightarrow & \exists y \in \{0,1\}^{p(|x|)}\ s.t.\ f(x\#y) \in A \\ & \Longrightarrow & \exists y \in \{0,1\}^{p(|x|)}\ s.t.\ x\#y \in B \end{array}$$

Similarly, $x \notin L \Longrightarrow \forall y \in \{0,1\}^{p(|x|)}$, $x\#y \notin B$. This proves that $L \in \ \exists \cdot B \subseteq \ \exists \cdot \mathcal{C}$.

The proof for (2) is similar. □

The class $\mathbf{BP}\cdot\oplus\mathbf{P}$ played an important role in Toda's well known proof [Tod91].

Corollary 4.8 $\mathbf{BP}\cdot\oplus\mathbf{P}$ *is gap-definable.*

Lemma 4.9 *Let $\mathcal{C}$ be any class containing $\{\emptyset, \Sigma^*\}$ and with the property that $L \in \mathcal{C}$ implies that $L\#0, L\#1 \in \mathcal{C}$. Then $L_1, L_2 \in \mathcal{C}$ implies that $L_1 \cup L_2 \in \mathbf{R}\cdot\mathrm{GapCl}(\mathcal{C})$.*

Proof: Let $B = (L_1\#0) \cup (L_2\#1)$. By Theorem 4.1, $B = (L_1\#0 \cap \Sigma^*\#0) \cup (L_2\#1 \cap \Sigma^*\#1) \in \mathrm{GapCl}(\mathcal{C})$ since $\Sigma^*\#0, \Sigma^*\#1 \in \mathbf{SPP}$ and $\Sigma^*\#0 \cap \Sigma^*\#1 = \emptyset$.

$$\begin{array}{rcl} x \in L_1 \cup L_2 & \Longrightarrow & x\#0 \in B \text{ or } x\#1 \in B \Longrightarrow Pr\{y \in \{0,1\} \mid x\#y \in B\} \geq 1/2 \\ x \notin L_1 \cup L_2 & \Longrightarrow & x\#0 \notin B \text{ and } x\#1 \notin B \Longrightarrow Pr\{y \in \{0,1\} \mid x\#y \in B\} = 0 \end{array}$$

Therefore, $L_1 \cup L_2 \in \mathbf{R}\cdot\mathrm{GapCl}(\mathcal{C})$. □

Corollary 4.10 *If $\mathcal{C}$ is closed under the $(\mathbf{R}\cdot)$-operation and contains $\{\emptyset, \Sigma^*\}$, and with the property that $L \in \mathcal{C}$ implies that $L\#0, L\#1 \in \mathcal{C}$, then $\mathcal{C}$ is gap-definable iff $\mathcal{C}$ is closed under union and closed under intersection with* $\mathbf{SPP}$.

Proof: If $\mathcal{C}$ is closed under the $(\mathbf{R}\cdot)$-operation, and gap-definable, then $\mathbf{R}\cdot\mathrm{GapCl}(\mathcal{C}) \subseteq \mathcal{C}$. By Lemma 4.9, it is closed under union. For any $L \in \mathcal{C}$ and $S \in \mathbf{SPP}, (L \cap S) = (L \cap S) \cup (\emptyset \cap \overline{S}) \in \mathcal{C}$ by Theorem 4.1. Conversely, if $\mathcal{C}$ is closed under union, and intersection with $\mathbf{SPP}$, then it is gap-definable by Theorem 4.1.
□

So far in this paper we have given simple characterizations of gap-definability in a broad range of circumstances, and showed that many classes not previously known to be gap-definable are indeed so. We have yet to give any result stating that gap-definability has structural consequences not related to the class **SPP** or other gap-definable classes. The following proposition is a step in that direction.

Proposition 4.11 *If $\mathcal{C}$ is a reasonable gap-definable class which is closed under m-reductions and complements, then $\mathcal{C}$ is closed under 1-tt-reductions.*

Proof: Suppose $L \in \mathcal{C}$ and $A \leq_{1-\mathrm{tt}} L$ via the polynomial-time function f. ($f(x) = \langle \alpha, y \rangle$ where $y \in \Sigma^*$, $\alpha \in \{T, F, id, \neg\}$ is one of the unary Boolean functions, and $x \in A$ iff $\alpha(y \in L)$.) For each $b \in \{T, F, id, \neg\}$, let $S_b \stackrel{\mathrm{df}}{=} \{x \mid (\exists y) f(x) = \langle b, y \rangle\}$. The set $S_b \in \mathbf{P} \subseteq \mathbf{SPP}$ for each b, and the four sets are pairwise disjoint. Let $B \stackrel{\mathrm{df}}{=} \{x \mid (\exists \alpha)(\exists y \in L) f(x) = \langle \alpha, y \rangle\}$. Since $B \leq_m L$, we have $B \in \mathcal{C}$, and $\overline{B} \in \mathcal{C}$ since $\mathcal{C}$ is closed under complements. It follows from the definition of 1-tt-reductions that $A = (\Sigma^* \cap S_T) \cup (\emptyset \cap S_F) \cup (B \cap S_{id}) \cup (\overline{B} \cap S_{\neg})$, and thus $A \in \mathcal{C}$ by Theorem 4.1. □

This result is nontrivial; for example, the class $\mathbf{NP} \cup \mathbf{coNP}$ is closed under m-reductions and complements but is not closed under 1-tt-reductions unless $\mathbf{NP} = \mathbf{coNP}$. Proposition 4.11 then implies that $\mathbf{NP} \cup \mathbf{coNP}$ is unlikely to be gap-definable.

Corollary 4.12 *If $\{\emptyset, \Sigma^*\} \subseteq \mathcal{C}$ and $\mathcal{C}$ is closed under m-reductions and complements, then* $\mathrm{GapCl}(\mathcal{C})$ *is closed under 1-tt-reductions.*

Proof: It was shown in [FFK94] that the GapCl operator preserves closure under m-reductions and closure under complements. □

We have seen that for many classes, having certain Boolean closure properties implies gap-definability. One may ask whether these Boolean closure properties are necessary. We will show, however, that not all gap-definable classes, not even all reasonable gap-definable classes, are closed under union (resp. intersection, complement).

Theorem 4.13

1. *There exists a reasonable gap-definable class which is not closed under union.*
2. *There exists a reasonable gap-definable class which is not closed under intersection.*
3. *There exists a reasonable gap-definable class which is not closed under complement.*

Proof:

1. We want to find L_1, L_2 s.t. $L_1 \cup L_2 \notin \mathrm{GapCl}(\{L_1, L_2, \emptyset, \Sigma^*\})$, i.e., $L_1 \cup L_2 \neq (L_1 \cap S_i) \cup (L_2 \cap S_j) \cup S_k$ for all disjoint $S_i, S_j, S_k \in \mathbf{SPP}$ (Theorem 3.1). Let $\mathbf{SPP} = \{S_1, S_2, \ldots\}$. Construct L_1 and L_2 as follows. For each pairwise disjoint triple $S_i, S_j, S_k \in \mathbf{SPP}$ pick $x = 0^{i+1}1^{j+1}0^{k+1}$.

   If $x \in S_k$ put $x \notin L_1 \cup L_2$

   If $x \in S_i$ put $x \in L_2 - L_1$

   If $x \in S_j$ put $x \in L_1 - L_2$

   If $x \notin S_i \cup S_j \cup S_k$ put $x \in L_1 - L_2$

   The set $L_1 \cup L_2$ is neither empty nor Σ^* nor contained in $\mathrm{GapCl}(\{L_1, L_2, \emptyset, \Sigma^*\})$. In fact, if we fix i, j, k, then $x = 0^{i+1}1^{j+1}0^{k+1}$ is either in $L_1 \cup L_2$ but not in $(L_1 \cap S_i) \cup (L_2 \cap S_j) \cup S_k$, or in S_k but not in $L_1 \cup L_2$.

2. For i, j, k, let $x = 0^{i+1}1^{j+1}0^{k+1}$,

   If $x \in S_k$ put $x \notin L_1 \cap L_2$

   If $x \in S_i$ put $x \in L_1 - L_2$

   If $x \in S_j$ put $x \in L_2 - L_1$

   If $x \notin S_i \cup S_j \cup S_k$ put $x \in L_1 \cap L_2$

A similar argument will show that $L_1 \cap L_2 \notin \mathrm{GapCl}(\{L_1, L_2, \emptyset, \Sigma^*\})$.

3. Let $L \notin$ **SPP**, then $L' \notin$ **SPP**, where L' is the complement of L. Then $L' \notin \mathrm{GapCl}(\{L, \emptyset, \Sigma^*\})$ for, otherwise, $L' = (L \cap S_1) \cup S_2$ for some disjoint $S_1, S_2 \in$ **SPP** by Theorem 3.1. Then we would have $L' = S_2$, a contradiction.

□

5 Separation by Gap

Although the questions whether **P** $\neq$ **PSPACE** or **SPP** $\neq$ **PSPACE** are still open, they are widely believed to be true. To better understand the relationships among them, it is helpful to look at the implications of these inequalities. In this section, we provide an equivalent condition for separation. For example, **SPP** $\neq$ **PSPACE** if and only if there is a "gap" between **PSPACE** and some gap-definable proper subclass closed under many-one reductions. In other words, classes like **PP** and **PSPACE** contain maximal gap-definable subclasses which are closed under many-one reductions, unless they equal **SPP**. By *maximal* $\mathcal{K}$-subclass we mean a proper $\mathcal{K}$-subclass such that there is no proper $\mathcal{K}$-subclass containing it.

Theorem 5.1 *Let* **SPP** *be properly contained in a class $\mathcal{D}$ which has a many-one complete set. Then there exists a maximal m-closed gap-definable class $\mathcal{M}$ such that* **SPP** $\subseteq \mathcal{M} \subset \mathcal{D}$.
(m-closed means closed under many-one reductions.)

Proof: Let $\mathcal{P}$ be the set of all classes $\mathcal{E}$ s.t.

1. **SPP** $\subseteq \mathcal{E} \subset \mathcal{D}$,
2. $\mathcal{E}$ is m-closed,
3. $\mathcal{E}$ is gap-definable.

The collection $\mathcal{P}$ is not empty since **SPP** $\in \mathcal{P}$. Let $\mathcal{A} \subseteq \mathcal{P}$ be a possibly uncountable chain under inclusion. We want to show that this chain has an upper bound in $\mathcal{P}$. Let $\mathcal{B} = \bigcup_{\mathcal{C} \in \mathcal{A}} \mathcal{C}$. It is clear that $\mathcal{B}$ is an upper bound of this chain. Now we show that $\mathcal{B}$ is in $\mathcal{P}$:

1. The class **SPP** $\subseteq \mathcal{B}$. Since $\mathcal{C} \subseteq \mathcal{D}$ for all $\mathcal{C} \in \mathcal{A}$, we have $\mathcal{B} \subseteq \mathcal{D}$. None of the $\mathcal{C}$'s contain a $\mathcal{D}$-complete set, for otherwise $\mathcal{C} = \mathcal{D}$ since $\mathcal{C}$ is m-closed. Thus $\mathcal{B} \neq \mathcal{D}$
2. $\mathcal{B}$ is m-closed: A language L many-one reduces to $\mathcal{B}$ implies that L many-one reduces to $\mathcal{C}$ for some $\mathcal{C} \in \mathcal{A}$. Since $\mathcal{C}$ is closed under many-one reductions, we have $L \in \mathcal{C} \subseteq \mathcal{B}$.
3. $\mathcal{B}$ is gap-definable:

$$
\begin{aligned}
L_1, L_2 \in \mathcal{B} &\implies L_1, L_2 \in \mathcal{C} \text{ for some } \mathcal{C} \in \mathcal{A} \\
&\implies (L_1 \cap S) \cup (L_2 \cap \overline{S}) \in \mathcal{C} \\
&\qquad \text{for all } S \in \mathbf{SPP} \text{ (Theorem 4.1)} \\
&\implies (L_1 \cap S) \cup (L_2 \cap \overline{S}) \in \mathcal{B} \\
&\qquad \text{for all } S \in \mathbf{SPP} \\
&\implies \mathcal{B} \text{ is gap-definable (Theorem 4.1).}
\end{aligned}
$$

Now applying Zorn's lemma, $\mathcal{P}$ has a maximal element. □

Corollary 5.2 *There exists a maximal m-closed gap-definable subclass of* **PSPACE** *unless* **PSPACE** = **SPP**.

Corollary 5.3 *There exists a maximal m-closed gap-definable subclass of* **PP** *unless* **SPP** = **PP**.

We further decompose the class $\mathcal{C}$ described above. We especially discuss the class **PSPACE**, but the result may be applied to other complexity classes with the necessary properties.

Definition 5.4 $\mathbf{PSPACE}_{\mathrm{COMP}}$ = *Class of all* **PSPACE** *complete sets.*
$\mathbf{PSPACE}_{\mathrm{INC}} = \mathbf{PSPACE} - \mathbf{PSPACE}_{\mathrm{COMP}}$.

First we note that $\mathbf{PSPACE}_{\mathrm{INC}}$ is a maximal m-closed subclass of **PSPACE**. In fact, if L is many-one reducible to $\mathbf{PSPACE}_{\mathrm{INC}}$, then $L \in \mathbf{PSPACE}$, and if we also have $L \in \mathbf{PSPACE}_{\mathrm{COMP}}$, then $\mathbf{PSPACE} \leq_m \mathbf{PSPACE}_{\mathrm{INC}}$, a contradiction. This means $\mathbf{PSPACE}_{\mathrm{INC}}$ is m-closed and there is no m-closed class $\mathcal{D}$ such that $\mathbf{PSPACE}_{\mathrm{INC}} \subset \mathcal{D} \subset \mathbf{PSPACE}$. Furthermore, $\mathbf{PSPACE}_{\mathrm{INC}}$ is the maximum m-closed subclass of **PSPACE**.

Suppose $\mathbf{SPP} \neq \mathbf{PSPACE}$. Let $\mathcal{M}$ be a maximal gap-definable m-closed subclass of **PSPACE**. From the above discussion, we have $\mathcal{M} \subseteq \mathbf{PSPACE}_{\mathrm{INC}}$. It is natural to ask whether $\mathcal{M} = \mathbf{PSPACE}_{\mathrm{INC}}$.

We will give a negative answer.

Lemma 5.5 $\mathrm{GapCl}(\mathbf{PSPACE}_{\mathrm{INC}}) = \mathbf{PSPACE}$.

Proof: We consider two cases. If $\mathbf{P} \neq \mathbf{PSPACE}$, then we apply a result of Ladner [Lad75] that there are $L_1, L_2 \in \mathbf{PSPACE}_{\mathrm{INC}}$ such that $L_1 \oplus L_2 \in \mathbf{PSPACE}_{\mathrm{COMP}}$. If $\mathbf{P} = \mathbf{PSPACE}$, then $L_1 = \emptyset$ and $L_2 = \Sigma^*$ are the only sets in $\mathbf{PSPACE}_{\mathrm{INC}}$, and $L_1 \oplus L_2 = \emptyset \oplus \Sigma^* \in \mathbf{PSPACE}_{\mathrm{COMP}}$. Therefore, in both cases, we have that there exist two sets $L_1, L_2 \in \mathbf{PSPACE}_{\mathrm{INC}}$ such that $L_1 \oplus L_2 \in \mathbf{PSPACE}_{\mathrm{COMP}}$. On the other hand,

$L_1 \oplus L_2 = 0L_1 \cup 1L_2 = (0L_1 \cap 0\Sigma^*) \cup (1L_2 \cap 1\Sigma^*) \in \mathrm{GapCl}(\mathbf{PSPACE}_{\mathrm{INC}})$.

The last inclusion is from Theorem 3.1.

Since $\mathbf{PSPACE}_{\mathrm{INC}}$ is m-closed, so is $\mathrm{GapCl}(\mathbf{PSPACE}_{\mathrm{INC}})$. Then the fact that $\mathrm{GapCl}(\mathbf{PSPACE}_{\mathrm{INC}})$ contains a **PSPACE**-complete set implies that it contains all of **PSPACE**. Since **PSPACE** is gap-definable (Corollary 4.6), the result follows. □

The lemma tells us two things. First, $\mathbf{PSPACE}_{\mathrm{INC}}$ is not gap-definable. Second, any gap-definable class containing $\mathbf{PSPACE}_{\mathrm{INC}}$ must contain all **PSPACE**-complete sets.

Now we return to the previous discussion. Assume **SPP** is separated from **PSPACE**. By Theorem 5.1, $\mathcal{M} = \mathbf{PSPACE}_{\mathrm{INC}}$ implies $\mathcal{M} = \mathbf{PSPACE}$ since $\mathcal{M}$ is gap-definable and therefore $\mathrm{GapCl}(\mathcal{M}) = \mathcal{M}$. This is impossible. Now we conclude that if **SPP** is separated from **PSPACE**, then there is a gap between $\mathcal{M}$ and $\mathbf{PSPACE}_{\mathrm{INC}}$. We further discuss the relationship between **SPP** and $\mathcal{M}$. Again, following Ladner's argument, we may find a set $A \in \mathbf{PSPACE} - \mathbf{SPP}$ such that $\mathbf{SPP}^A \subset \mathbf{PSPACE}$. The class $\mathbf{SPP}^A$ is closed under Turing reductions and joins. Therefore $\mathbf{SPP}^A$ is closed under union and intersection. By Corollary 4.2, $\mathbf{SPP}^A$ is gap-definable. It is also clear that $\mathbf{SPP}^A$ is m-closed. This means that **SPP** cannot be a maximal m-closed gap-definable class in **PSPACE**. Thus we also have the separation of **SPP** from $\mathcal{M}$. We summarize these results as the following

Theorem 5.6 *Unless* $\mathbf{PSPACE} = \mathbf{SPP}$*, we have*

$$\mathbf{SPP} \subset \mathcal{M} \subset \mathbf{PSPACE}_{\mathrm{INC}} \subset \mathbf{PSPACE},$$

where $\mathcal{M}$ *is defined as above.*

Using the same techniques one can prove a similar result replacing **SPP** by **PP**. Figure 1 illustrates a layout of **PSPACE** if **PP** is not equal to **PSPACE**.

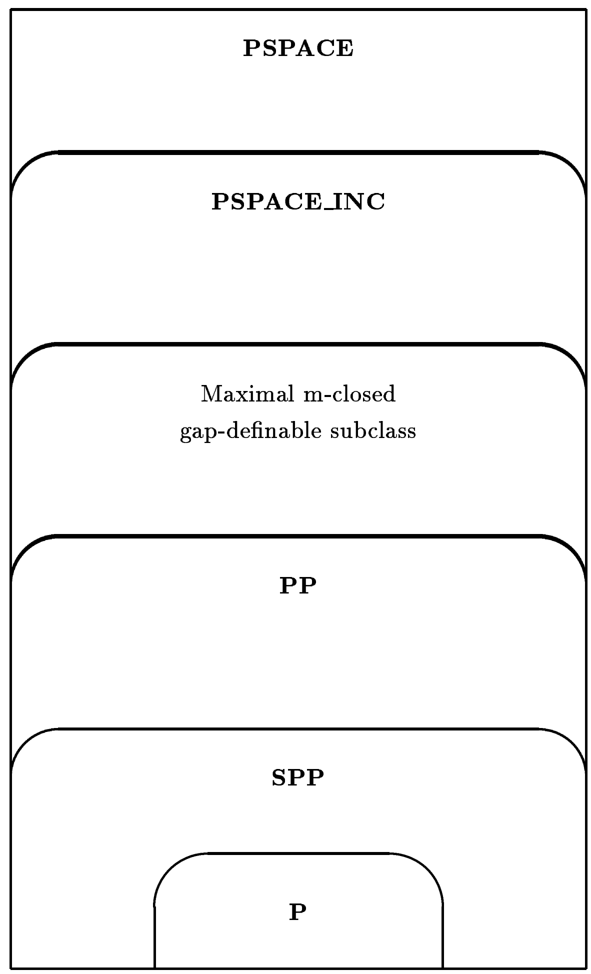

Figure 1: A Layout of PSPACE.
The thick lines indicate the real separations if $\mathbf{PP} \neq \mathbf{PSPACE}$.

References

[BG92] R. Beigel and J. Gill. Counting classes: Thresholds, parity, mods, and fewness. *Theoretical Computer Science*, 103:3–23, 1992.

[BRS95] R. Beigel, N. Reingold, and D. Spielman. PP is closed under intersection. *Journal of Computer and System Sciences*, 50(2):191–202, 1995.

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

[CH90] J. Cai and L. Hemachandra. On the power of parity polynomial time. *Mathematical Systems Theory*, 23(2):95–106, 1990.

[FFK94] S. Fenner, L. Fortnow, and S. Kurtz. Gap-definable counting classes. *Journal of Computer and System Sciences*, 48(1):116–148, 1994.

[Gil77] J. Gill. Computational complexity of probabilistic complexity classes. *SIAM Journal on Computing*, 6:675–695, 1977.

[GKR$^+$95] F. Green, J. Köbler, K. Regan, T. Schwentick, and J. Torán. The power of the middle bit of a #P function. *Journal of Computer and System Sciences*, 50(3):456–467, 1995.

[Her90] U. Hertrampf. Relations among MOD classes (note). *Theoretical Computer Science*, 74:325–328, 1990.

[Lad75] R. Ladner. On the structure of polynomial time reducibility. *Journal of the ACM*, 22:155–171, 1975.

[Sim75] J. Simon. *On Some Central Problems in Computational Complexity.* PhD thesis, Cornell University, 1975. Technical Report TR 75-224.

[TO92] S. Toda and M. Ogiwara. Counting classes are at least as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 21(2):316–328, 1992.

[Tod91] S. Toda. PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877, 1991.

[Val79] L. Valiant. The complexity of computing the permanent. *Theoretical Computer Science*, 8:189–201, 1979.

[Wag86] K. Wagner. The complexity of combinatorial problems with succinct input representation. *Acta Informatica*, 23:325–356, 1986.