

## The Intrinsic Complexity of Learning: A Survey

**Sanjay Jain\***

*School of Computing*

*National University of Singapore*

*Singapore 119260, Republic of Singapore*

*sanjay@comp.nus.edu.sg*

---

**Abstract.** The theory of learning in the limit has been a focus of study by several researchers over the last three decades. There have been several suggestions on how to measure the complexity or hardness of learning. In this paper we survey the work done in one specific such measure, called intrinsic complexity of learning. We will be mostly concentrating on learning languages, with only a brief look at function learning.

### 1. Introduction

Consider the identification of formal languages from positive data. A machine is fed all the strings and nonstrings of a language  $L$ , in any order, one string at a time. The machine, as it is receiving strings of  $L$ , outputs a sequence of grammars. The machine is said to identify  $L$  just in case the sequence of grammars converges to a grammar for  $L$ . This is essentially the paradigm of identification in the limit (called **TxtEx**-identification) introduced by Gold [11]. Identification of total functions from their graphs can be modeled similarly with the machine receiving as input elements of the graph of the function. Note that in function learning, a machine can deduce negative data, since presence of  $(x, y)$  in the input implies that  $(x, z)$  is not in the input for any  $z \neq y$ .

The theory of learning in the limit has been a focus of study by several researchers over the last three decades. We direct the reader to [15] for an introduction to the area. There have been several suggestions on how to measure the complexity or hardness of learning. Some of these are:

a) counting the number of *mind changes* [2, 6, 24] made by the learner before it converges to a final hypothesis;

---

\*Address for correspondence: School of Computing, National University of Singapore, Singapore 119260, Republic of Singapore

- b) measuring the time taken by the machine (“area under the curve”) before converging to the final hypothesis [7];
- c) measuring the amount of (so-called *long-term*) memory the learner uses [21, 23];
- d) number of examples needed before convergence [30]; and
- e) intrinsic complexity of learning [9, 10, 17].

The aim of this paper is to survey the work done in intrinsic complexity of learning. As mentioned above, two models of learning are usually considered in the literature: language learning from texts (positive data) and function learning from graphs. In some cases, language learning from both positive and negative data (informants) is also considered. In this survey, we will be concentrating mostly on the intrinsic complexity of language identification from texts, and only briefly look at function identification from graphs, and language identification from informants.

The origins of intrinsic complexity of learning date back to a paper by Freivalds [9] and were first developed for function learning by Freivalds, Kinber and Smith [10]. Jain and Sharma [17] first studied intrinsic complexity for language learning. We illustrate the notion using learning of some commonly considered classes of languages.

The following discussion is from [17]. Consider the following three collections of languages over  $N$ , the set of natural numbers.

$$\begin{aligned} \text{SINGLE} &= \{L \mid \text{card}(L) = 1\} \\ \text{COINIT} &= \{L \mid (\exists n)[L = \{x \mid x \geq n\}]\} \\ \text{FIN} &= \{L \mid \text{cardinality of } L \text{ is finite}\} \end{aligned}$$

So, SINGLE is the collection of all singleton languages, COINIT is the collection of languages that contain all natural numbers except a finite initial segment, and FIN is the collection of all finite languages. Clearly, each of these three classes is identifiable in the limit from only positive data. For example, a machine  $M_1$  that upon encountering the first data element, say  $n$ , keeps on emitting a grammar for the singleton language  $\{n\}$  identifies SINGLE. A machine  $M_2$  that, at any given time, finds the minimum element among the data seen so far, say  $n$ , and emits a grammar for the language  $\{x \mid x \geq n\}$  can easily be seen to identify COINIT. Similarly, a machine  $M_3$  that continually outputs a grammar for the finite set of data seen so far identifies FIN.

Now, although all three of these classes are identifiable, it can be argued that they present learning problems of varying difficulty. One way to look at the difficulty is to ask the question, “At what stage in the processing of the data can a learning machine confirm its success?” In the case of SINGLE, the machine can be confident of success as soon as it encounters the first data element. In the case of COINIT, the machine cannot always be sure that it has identified the language. However, at any stage after it has seen the first data element, the machine can provide an upper bound on the number of mind changes that the machine will make before converging to a correct grammar. For example, if at some stage the minimum element seen is  $m$ , then  $M_2$  will make no more than  $m$  mind changes because it changes its mind only if a smaller element appears. In the case of FIN, the learning machine can neither be confident about its success nor can it, at any stage, provide an upper bound on the number of further mind changes that it may have to undergo before it is rewarded with success. Clearly, these three collections of languages pose learning problems of varying difficulty where SINGLE appears to be the least difficult to learn and FIN is seen to be the most difficult to learn with COINIT appearing to be of intermediate difficulty.

We next present an informal description of the reductions that are central to our analysis of the intrinsic complexity of language learning. To facilitate our discussion, we first present some technical notions about language learning.

Informally, a *text* for a language  $L$  is just an infinite sequence of elements, with possible repetitions, of all and only the elements of  $L$ . A text for  $L$  is thus an abstraction of the presentation of positive data about  $L$ . A learning machine is essentially an algorithmic device. Elements of a text are sequentially fed to a learning machine one element at a time. The learning machine, as it receives elements of the text, outputs an infinite sequence of grammars. Several criteria for the learning machine to be successful on a text have been proposed. In the present paper we will concern ourselves with Gold's [11] criterion of identification in the limit (referred to as **TxtEx**-identification). A sequence of grammars,  $G = g_0, g_1, \dots$ , is said to *converge to*  $g$  just in case, for all but finitely many  $n$ ,  $g_n = g$ . We say that the sequence of grammars,  $G = g_0, g_1, \dots$ , converges just in case there exists a  $g$  such that  $G$  converges to  $g$ ; if no such  $g$  exists, then we say that the sequence  $G$  diverges. We say that  $\mathbf{M}$  converges on  $T$  (to  $g$ ), if the sequence of grammars emitted by  $\mathbf{M}$  on  $T$  converges (to  $g$ ). If the sequence of grammars emitted by the learning machine converges to a correct grammar for the language whose text is fed to the machine, then the machine is said to **TxtEx**-*identify* the text. A machine is said to **TxtEx**-*identify* a language just in case it **TxtEx**-identifies each text for the language.

It is also useful to call an infinite sequence of grammars,  $g_0, g_1, g_2, \dots$ , **TxtEx**-admissible for a text  $T$  just in case the sequence of grammars converges to a single correct grammar for the language whose text is  $T$ .

Our reductions are based on the idea that for a collection of languages  $\mathcal{L}$  to be reducible to  $\mathcal{L}'$ , we should be able to transform texts  $T$  for languages in  $\mathcal{L}$  to texts  $T'$  for languages in  $\mathcal{L}'$  and further transform **TxtEx**-admissible sequences for  $T'$  into **TxtEx**-admissible sequences for  $T$ . This is achieved with the help of two enumeration operators. Informally, enumeration operators are algorithmic devices that map infinite sequences of objects (for example, texts and infinite sequences of grammars) into infinite sequences of objects. The first operator,  $\Theta$ , transforms texts for languages in  $\mathcal{L}$  into texts for languages in  $\mathcal{L}'$ . The second operator,  $\Psi$ , behaves as follows: if  $\Theta$  transforms a text  $T$  for some language in  $\mathcal{L}$  into text  $T'$  (for some language in  $\mathcal{L}'$ ), then  $\Psi$  transforms **TxtEx**-admissible sequences for  $T'$  into **TxtEx**-admissible sequences for  $T$ .

To see that the above satisfies the intuitive notion of reduction consider collections  $\mathcal{L}$  and  $\mathcal{L}'$  such that  $\mathcal{L}$  is reducible to  $\mathcal{L}'$ . We now argue that if  $\mathcal{L}'$  is identifiable then so is  $\mathcal{L}$ .

Let  $\mathbf{M}'$  **TxtEx**-identify  $\mathcal{L}'$ . Let enumeration operators  $\Theta$  and  $\Psi$  witness the reduction of  $\mathcal{L}$  to  $\mathcal{L}'$ . We now describe a machine  $\mathbf{M}$  that **TxtEx**-identifies  $\mathcal{L}$ .  $\mathbf{M}$ , upon being fed a text  $T$  for some language  $L \in \mathcal{L}$ , uses  $\Theta$  to construct a text  $T'$  for a language in  $\mathcal{L}'$ . It then simulates machine  $\mathbf{M}'$  on text  $T'$  and feeds conjectures of  $\mathbf{M}'$  to the operator  $\Psi$  to produce its conjectures. It is easy to verify that the properties of  $\Theta$ ,  $\Psi$ , and  $\mathbf{M}'$  guarantee the success of  $\mathbf{M}$  on each text for each language in  $\mathcal{L}$ .

We also consider a stronger notion of reduction than the one discussed above. The reader should note that in the above reduction, different texts for the same language may be transformed into texts for different languages by  $\Theta$ . If we further require that  $\Theta$  is such that it transforms every text for a language into texts for some unique language then we have a stronger notion of reduction. In the context of function learning these two notions of reduction are the same [10]. However, in the context of language identification, from texts, this stronger notion of reduction turns out to be different from its weaker counterpart.

In this paper we will survey several results from the literature on intrinsic complexity of learning, mainly concentrating on structural results, complete classes, and some characterizations. We will only be giving a few simple sample proofs. We direct the reader to respective papers cited for the proofs.

We now proceed formally.

## 2. Notation and Preliminaries

Any unexplained recursion-theoretic notation is from [28]. The symbol  $N$  denotes the set of natural numbers,  $\{0, 1, 2, 3, \dots\}$ . Symbols  $\emptyset, \subseteq, \subset, \supseteq, \supset$  denote empty set, subset, proper subset, superset, and proper superset, respectively.  $D_0, D_1, \dots$ , denotes a canonical recursive indexing of all the finite sets [28, Page 70]. We assume that if  $D_i \subseteq D_j$  then  $i \leq j$  (the canonical indexing defined by Rogers [28] satisfies this property). Cardinality of a set  $S$  is denoted by  $\text{card}(S)$ . The maximum and minimum of a set are denoted by  $\max(\cdot), \min(\cdot)$ , respectively, where  $\max(\emptyset) = 0$  and  $\min(\emptyset) = \infty$ .

We let  $\langle \cdot, \cdot \rangle$  stand for an arbitrary, computable, bijective mapping from  $N \times N$  onto  $N$  [28]. We assume without loss of generality that  $\langle \cdot, \cdot \rangle$  is monotonically increasing in both of its arguments. We define  $\pi_1(\langle x, y \rangle) = x$  and  $\pi_2(\langle x, y \rangle) = y$ .  $\langle \cdot, \cdot \rangle$  can be extended to  $n$ -tuples in a natural way (including  $n = 1$ , where  $\langle x \rangle$  may be taken to be  $x$ ). Projection functions  $\pi_1, \dots, \pi_n$  corresponding to  $n$ -tuples can be defined similarly (where the tuple size would be clear from context). Due to the above isomorphism between  $N^k$  and  $N$ , we often identify the tuple  $(x_1, \dots, x_n)$  with  $\langle x_1, \dots, x_n \rangle$ .

By  $\varphi$  we denote a fixed *acceptable* programming system for the partial computable functions mapping  $N$  to  $N$  [28, 25]. By  $\varphi_i$  we denote the partial computable function computed by the program with number  $i$  in the  $\varphi$ -system. Symbol  $\mathcal{R}$  denotes the set of all recursive functions, that is total computable functions. By  $\Phi$  we denote an arbitrary fixed Blum complexity measure [3, 12] for the  $\varphi$ -system. By  $W_i$  we denote  $\text{domain}(\varphi_i)$ .  $W_i$  is, then, the r.e. set/language ( $\subseteq N$ ) accepted (or equivalently, generated) by the  $\varphi$ -program  $i$ . We also say that  $i$  is a grammar for  $W_i$ . Symbol  $\mathcal{E}$  will denote the set of all r.e. languages. Symbol  $L$ , with or without decorations, ranges over  $\mathcal{E}$ . By  $\overline{L}$ , we denote the complement of  $L$ , that is  $N - L$ . Symbol  $\mathcal{L}$ , with or without decorations, ranges over subsets of  $\mathcal{E}$ . We denote by  $W_{i,s}$  the set  $\{x \leq s \mid \Phi_i(x) < s\}$ .

A class  $\mathcal{L} \subseteq \mathcal{E}$  is said to be recursively enumerable (r.e.) [28], iff  $\mathcal{L} = \emptyset$  or there exists a recursive function  $f$  such that  $\mathcal{L} = \{W_{f(i)} \mid i \in N\}$ . In this latter case we say that  $W_{f(0)}, W_{f(1)}, \dots$  is a recursive enumeration of  $\mathcal{L}$ .  $\mathcal{L}$  is said to be 1-1 enumerable iff (i)  $\mathcal{L}$  is finite or (ii) there exists a recursive function  $f$  such that  $\mathcal{L} = \{W_{f(i)} \mid i \in N\}$  and  $W_{f(i)} \neq W_{f(j)}$ , if  $i \neq j$ . In this latter case we say that  $W_{f(0)}, W_{f(1)}, \dots$  is a 1-1 recursive enumeration of  $\mathcal{L}$ .

A partial function  $F$  from  $N$  to  $N$  is said to be partial limit recursive, iff there exists a recursive function  $f$  from  $N \times N$  to  $N$  such that for all  $x$ ,  $F(x) = \lim_{y \rightarrow \infty} f(x, y)$ . Here if  $F(x)$  is not defined then  $\lim_{y \rightarrow \infty} f(x, y)$ , must also be undefined. A partial limit recursive function  $F$  is called (total) limit recursive function, if  $F$  is total.  $\downarrow$  denotes defined or converges.  $\uparrow$  denotes undefined or diverges.

## 3. Language Identification

We now present concepts from language learning theory. The next definition introduces the concept of a *sequence* of data.

**Definition 3.1.** (a) A *sequence*  $\sigma$  is a mapping from an initial segment of  $N$  into  $(N \cup \{\#\})$ . The empty sequence is denoted by  $\Lambda$ .

(b) The *content* of a sequence  $\sigma$ , denoted  $\text{content}(\sigma)$ , is the set of natural numbers in the range of  $\sigma$ .

(c) The *length* of  $\sigma$ , denoted by  $|\sigma|$ , is the number of elements in  $\sigma$ . So,  $|\Lambda| = 0$ .

(d) For  $n \leq |\sigma|$ , the initial sequence of  $\sigma$  of length  $n$  is denoted by  $\sigma[n]$ . So,  $\sigma[0]$  is  $\Lambda$ .

Intuitively,  $\#$ 's represent pauses in the presentation of data. We let  $\sigma$ ,  $\tau$ , and  $\gamma$ , with or without decorations, range over finite sequences. We denote the sequence formed by the concatenation of  $\tau$  at the end of  $\sigma$  by  $\sigma \diamond \tau$ . Sometimes we abuse the notation and use  $\sigma \diamond x$  to denote the concatenation of sequence  $\sigma$  and the sequence of length 1 which contains the element  $x$ . SEQ denotes the set of all finite sequences.

**Definition 3.2.** [11] (a) A *text*  $T$  for a language  $L$  is a mapping from  $N$  into  $(N \cup \{\#\})$  such that  $L$  is the set of natural numbers in the range of  $T$ .

(b) The *content* of a text  $T$ , denoted by  $\text{content}(T)$ , is the set of natural numbers in the range of  $T$ ; that is, the language which  $T$  is a text for.

(c)  $T[n]$  denotes the finite initial sequence of  $T$  with length  $n$ .

We let  $T$ , with or without decorations, range over texts. We let  $\mathcal{T}$  range over sets of texts.

A class  $\mathcal{T}$  of texts is said to be r.e. if there exists a recursive function  $f$ , and a sequence  $T_0, T_1, \dots$  of texts such that  $\mathcal{T} = \{T_i \mid i \in N\}$ , and, for all  $i, x$ ,  $T_i(x) = f(i, x)$ .

**Definition 3.3.** [11] A *language learning machine* is an algorithmic device which computes a mapping from SEQ into  $N$ .

We let  $\mathbf{M}$ , with or without decorations, range over learning machines.  $\mathbf{M}(T[n])$  is interpreted as the grammar (index for an accepting program) conjectured by the learning machine  $\mathbf{M}$  on the initial sequence  $T[n]$ . We say that  $\mathbf{M}$  converges on  $T$  to  $i$ , (written  $\mathbf{M}(T) \downarrow = i$ ) if  $(\forall^\infty n)[\mathbf{M}(T[n]) = i]$ .

There are several criteria for a learning machine to be successful on a language. Below we define identification in the limit introduced by Gold [11].

**Definition 3.4.** [11]

(a)  $\mathbf{M}$  **TxtEx**-identifies a text  $T$  just in case  $(\exists i \mid W_i = \text{content}(T)) (\forall^\infty n)[\mathbf{M}(T[n]) = i]$ .

(b)  $\mathbf{M}$  **TxtEx**-identifies an r.e. language  $L$  (written:  $L \in \mathbf{TxtEx}(\mathbf{M})$ ) just in case  $\mathbf{M}$  **TxtEx**-identifies each text for  $L$ .

(c)  $\mathbf{M}$  **TxtEx**-identifies a class  $\mathcal{L}$  of r.e. languages (written:  $\mathcal{L} \subseteq \mathbf{TxtEx}(\mathbf{M})$ ) just in case  $\mathbf{M}$  **TxtEx**-identifies each language from  $\mathcal{L}$ .

(d)  $\mathbf{TxtEx} = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{TxtEx}(\mathbf{M})]\}$ .

Other criteria of success are finite identification [11], behaviorally correct identification [8, 27, 5], and vacillatory identification [27, 4]. In the present survey, we only discuss results about **TxtEx**-identification (sometimes with anomalies; see Section 12).

## 4. Weak and Strong Reductions

We first present some technical machinery.

We write  $\sigma \subseteq \tau$  if  $\sigma$  is an initial segment of  $\tau$ , and  $\sigma \subset \tau$  if  $\sigma$  is a proper initial segment of  $\tau$ . Likewise, we write  $\sigma \subset T$  if  $\sigma$  is an initial finite sequence of text  $T$ . Let finite sequences  $\sigma^0, \sigma^1, \sigma^2, \dots$  be given such that  $\sigma^0 \subseteq \sigma^1 \subseteq \sigma^2 \subseteq \dots$  and  $\lim_{i \rightarrow \infty} |\sigma^i| = \infty$ . Then there is a unique text  $T$  such that for all  $n \in \mathbb{N}$ ,  $\sigma^n = T[|\sigma^n|]$ . This text is denoted by  $\bigcup_n \sigma^n$ . Let  $\mathbf{T}$  denote the set of all texts, that is, the set of all infinite sequences over  $N \cup \{\#\}$ .

We define an *enumeration operator* (or just operator),  $\Theta$ , to be an algorithmic mapping from SEQ into SEQ such that for all  $\sigma, \tau \in \text{SEQ}$ , if  $\sigma \subseteq \tau$ , then  $\Theta(\sigma) \subseteq \Theta(\tau)$ . We further assume that for all texts  $T$ ,  $\lim_{n \rightarrow \infty} |\Theta(T[n])| = \infty$ . By extension, we think of  $\Theta$  as also defining a mapping from  $\mathbf{T}$  into  $\mathbf{T}$  such that  $\Theta(T) = \bigcup_n \Theta(T[n])$ .

A final notation about the operator  $\Theta$ .  $\Theta(L) = \{\text{content}(\Theta(T)) \mid T \text{ is a text for } L\}$ . The reader should note the overloading of notation because the type of the argument to  $\Theta$  could be a sequence, a text, or a language; it will be clear from the context which usage is intended.

We let  $\Theta(\mathcal{T}) = \{\Theta(T) \mid T \in \mathcal{T}\}$ , and  $\Theta(\mathcal{L}) = \bigcup_{L \in \mathcal{L}} \Theta(L)$ .

We also need the notion of an infinite sequence of grammars. We let  $\alpha$ , with or without decorations, range over infinite sequences of grammars. From the discussion in the previous section it is clear that infinite sequences of grammars are essentially infinite sequences over  $N$ . Hence, we adopt the machinery defined for sequences and texts over to finite sequences of grammars and infinite sequences of grammars. So, if  $\alpha = i_0, i_1, i_2, i_3, \dots$ , then  $\alpha[3]$  denotes the sequence  $i_0, i_1, i_2$ , and  $\alpha(3)$  is  $i_3$ . Furthermore, we say that  $\alpha$  converges to  $i$  if there exists an  $n$  such that, for all  $n' \geq n$ ,  $i_{n'} = i$ .

Let  $\mathbf{I}$  be any criterion for language identification from texts, for example  $\mathbf{I} = \mathbf{TextEx}$ . We say that an infinite sequence  $\alpha$  of grammars is  $\mathbf{I}$ -admissible for text  $T$  just in case  $\alpha$  witnesses  $\mathbf{I}$ -identification of text  $T$ . So, if  $\alpha = i_0, i_1, i_2, \dots$  is a  $\mathbf{TextEx}$ -admissible sequence for  $T$ , then  $\alpha$  converges to some  $i$  such that  $W_i = \text{content}(T)$ ; that is, the limit  $i$  of the sequence  $\alpha$  is a grammar for the language  $\text{content}(T)$ .

We now formally introduce our reductions.

**Definition 4.1.** [17] Let  $\mathcal{L}_1 \subseteq \mathcal{E}$  and  $\mathcal{L}_2 \subseteq \mathcal{E}$  be given. Let identification criteria  $\mathbf{I}_1$  and  $\mathbf{I}_2$  be given. Let  $\mathcal{T}_1 = \{T \mid T \text{ is a text for } L \in \mathcal{L}_1\}$ . Let  $\mathcal{T}_2 = \{T \mid T \text{ is a text for } L \in \mathcal{L}_2\}$ . We say that  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}_1, \mathbf{I}_2} \mathcal{L}_2$  just in case there exist operators  $\Theta$  and  $\Psi$  such that for all  $T \in \mathcal{T}_1$  and for all infinite sequences  $\alpha$  of grammars the following hold:

(a)  $\Theta(T) \in \mathcal{T}_2$  and

(b) if  $\alpha$  is an  $\mathbf{I}_2$ -admissible sequence for  $\Theta(T)$ , then  $\Psi(\alpha)$  is an  $\mathbf{I}_1$ -admissible sequence for  $T$ .

We say that  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  iff  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}, \mathbf{I}} \mathcal{L}_2$ . We say that  $\mathcal{L}_1 \equiv_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  iff  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  and  $\mathcal{L}_2 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_1$ .

Intuitively,  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  just in case there exists an operator  $\Theta$  that transforms texts for languages in  $\mathcal{L}_1$  into texts for languages in  $\mathcal{L}_2$  and there exists another operator  $\Psi$  that behaves as follows: if  $\Theta$  transforms text  $T$  (for a language in  $\mathcal{L}_1$ ) to text  $T'$  (for a language in  $\mathcal{L}_2$ ), then  $\Psi$  transforms  $\mathbf{I}$ -admissible sequences for  $T'$  into  $\mathbf{I}$ -admissible sequences for  $T$ . Thus, informally, the operator  $\Psi$  has “to work” only on  $\mathbf{I}$ -admissible sequences for such texts  $T'$ . In other words, if  $\alpha$  is a sequence of grammars which is *not*  $\mathbf{I}$ -admissible for any text  $T'$  in  $\{\Theta(T) \mid \text{content}(T) \in \mathcal{L}_1\}$ , then  $\Psi(\alpha)$  can be defined *arbitrarily*.

Intuitively, for many commonly studied criteria of inference, such as  $\mathbf{I} = \mathbf{TextEx}$ , if  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  then the problem of identifying  $\mathcal{L}_2$  in the sense of  $\mathbf{I}$  is at least as hard as the problem of identifying  $\mathcal{L}_1$  in

the sense of  $\mathbf{I}$ , since the solvability of the former problem implies the solvability of the latter one. That is, given any machine  $\mathbf{M}_2$  which  $\mathbf{I}$ -identifies  $\mathcal{L}_2$ , it is easy to construct a machine  $\mathbf{M}_1$  which  $\mathbf{I}$ -identifies  $\mathcal{L}_1$ . To see this for  $\mathbf{I} = \mathbf{TxtEx}$ , suppose  $\Theta$  and  $\Psi$  witness  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$ .  $\mathbf{M}_1(T)$ , for a text  $T$ , is defined as follows. Let  $p_n = \mathbf{M}_2(\Theta(T)[n])$ , and  $\alpha = p_0, p_1, \dots$ . Let  $\alpha' = \Psi(\alpha) = p'_0, p'_1, \dots$ . Then let  $\mathbf{M}_1(T) = \lim_{n \rightarrow \infty} p'_n$ . Consequently,  $\mathcal{L}_2$  may be considered as a “hardest” problem for  $\mathbf{I}$ -identification if for *all* classes  $\mathcal{L}_1 \in \mathbf{I}$ ,  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  holds. If  $\mathcal{L}_2$  itself belongs to  $\mathbf{I}$ , then  $\mathcal{L}_2$  is said to be *complete*. We now formally define these notions of hardness and completeness for the above reduction.

**Definition 4.2.** [17] Let  $\mathbf{I}$  be an identification criterion. Let  $\mathcal{L} \subseteq \mathcal{E}$  be given.

- (a) If for all  $\mathcal{L}' \in \mathbf{I}$ ,  $\mathcal{L}' \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}$ , then  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\mathbf{I}}$ -hard.
- (b) If  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\mathbf{I}}$ -hard and  $\mathcal{L} \in \mathbf{I}$ , then  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\mathbf{I}}$ -complete.

It should be noted that if  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}_2$  is witnessed by  $\Theta$  and  $\Psi$ , then there is no requirement that  $\Theta$  maps all texts for each language in  $\mathcal{L}_1$  into texts for a *unique* language in  $\mathcal{L}_2$ . If we further place such a constraint on  $\Theta$ , we get the following stronger notion.

**Definition 4.3.** [17] Let  $\mathcal{L}_1 \subseteq \mathcal{E}$  and  $\mathcal{L}_2 \subseteq \mathcal{E}$  be given. We say that  $\mathcal{L}_1 \leq_{\text{strong}}^{\mathbf{I}_1, \mathbf{I}_2} \mathcal{L}_2$  just in case there exist operators  $\Theta, \Psi$  witnessing that  $\mathcal{L}_1 \leq_{\text{weak}}^{\mathbf{I}_1, \mathbf{I}_2} \mathcal{L}_2$ , and for all  $L_1 \in \mathcal{L}_1$ , there exists an  $L_2 \in \mathcal{L}_2$ , such that  $(\forall \text{ texts } T \text{ for } L_1)[\Theta(T) \text{ is a text for } L_2]$ .

We say that  $\mathcal{L}_1 \leq_{\text{strong}}^{\mathbf{I}} \mathcal{L}_2$  iff  $\mathcal{L}_1 \leq_{\text{strong}}^{\mathbf{I}, \mathbf{I}} \mathcal{L}_2$ . We say that  $\mathcal{L}_1 \equiv_{\text{strong}}^{\mathbf{I}} \mathcal{L}_2$  iff  $\mathcal{L}_1 \leq_{\text{strong}}^{\mathbf{I}} \mathcal{L}_2$  and  $\mathcal{L}_2 \leq_{\text{strong}}^{\mathbf{I}} \mathcal{L}_1$ .

We can similarly define  $\leq_{\text{strong}}^{\mathbf{I}}$ -hardness and  $\leq_{\text{strong}}^{\mathbf{I}}$ -completeness.

**Proposition 4.1.** ([17])  $\leq_{\text{weak}}^{\mathbf{TxtEx}}$ ,  $\leq_{\text{strong}}^{\mathbf{TxtEx}}$  are reflexive and transitive.

The above proposition holds for most natural learning criteria. It is also easy to verify the next proposition stating that strong reducibility implies weak reducibility.

**Proposition 4.2.** [17] Let  $\mathcal{L} \subseteq \mathcal{E}$  and  $\mathcal{L}' \subseteq \mathcal{E}$  be given. Let  $\mathbf{I}$  be an identification criterion. Then  $\mathcal{L} \leq_{\text{strong}}^{\mathbf{I}} \mathcal{L}' \Rightarrow \mathcal{L} \leq_{\text{weak}}^{\mathbf{I}} \mathcal{L}'$ .

## 5. Some Properties of Reductions

In this section we will present some lemmas which illustrate some of the important properties of reductions. These properties are mainly constraints which have to be satisfied by  $\Theta$  witnessing a reduction from some class to another. These properties are often useful in showing nonreducibility results.

**Lemma 5.1.** [19] Suppose  $\mathcal{L} \subseteq \mathcal{E}$ ,  $\mathcal{L}' \subseteq \mathcal{E}$  and  $\mathcal{L} \leq_{\text{weak}}^{\mathbf{TxtEx}} \mathcal{L}'$  as witnessed by  $\Theta$  and  $\Psi$ . Then

- (a)  $(\forall L \in \mathcal{L})[\Theta(L) \subseteq \mathcal{L}']$ ;
- (b)  $(\forall L, L' \in \mathcal{L})[L \neq L' \Rightarrow \Theta(L) \cap \Theta(L') = \emptyset]$ .

Part (b) of the above lemma essentially says that  $\Theta$  (witnessing a  $\leq_{\text{weak/strong}}^{\mathbf{TxtEx}}$  reduction from  $\mathcal{L}$  to  $\mathcal{L}'$ ) cannot map (texts for) two distinct languages in  $\mathcal{L}$  to (texts for) the same language in  $\mathcal{L}'$ .

**Lemma 5.2.** [19] Suppose  $\Theta$  is an enumeration operator.

- (a) Suppose  $L_1 \subseteq L_2 \subseteq N$ , and  $L'_1 \in \Theta(L_1)$ . Then for every finite subset  $S$  of  $L'_1$ , there exists an  $L'_2 \in \Theta(L_2)$  such that  $S \subseteq L'_2$ .
- (b) Suppose  $L_1 \subseteq L_2 \subseteq N$ . Suppose further that  $\Theta(L_1)$  consists only of finite languages. Then for all  $L'_1 \in \Theta(L_1)$ , there exists an  $L'_2 \in \Theta(L_2)$  such that  $L'_1 \subseteq L'_2$ .

Part (a) of the following Corollary essentially says that “subset” relation is preserved in strong reduction. That is, if  $\Theta$  (along with  $\Psi$ ) witnesses that  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ , then  $L_1 \subseteq L_2$  implies  $\Theta(L_1) \subseteq \Theta(L_2)$ , for all  $L_1, L_2 \in \mathcal{L}$ .

A weaker version of above holds for weak reductions (part (b) of the following corollary).

**Corollary 5.1.** [19] Let  $\mathcal{L} \subseteq \mathcal{E}, \mathcal{L}' \subseteq \mathcal{E}$ .

- (a) Suppose  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$  as witnessed by  $\Theta$  and  $\Psi$ . Suppose  $L_1, L_2 \in \mathcal{L}$  and  $L_1 \subseteq L_2$ . Let  $S_1, S_2$  be such that  $\Theta(L_1) = \{S_1\}$  and  $\Theta(L_2) = \{S_2\}$ . Then  $S_1 \subseteq S_2$ .
- (b) Suppose  $\mathcal{L} \leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}'$  as witnessed by  $\Theta$  and  $\Psi$ . Suppose  $L_1, L_2 \in \mathcal{L}$  and  $L_1 \subseteq L_2$ . Further suppose that  $\mathcal{L}'$  consists only of finite languages. Then, for every  $S_1 \in \Theta(L_1)$ , there exists an  $S_2 \in \Theta(L_2)$  such that  $S_1 \subseteq S_2$ .

The above result can essentially be used to show that several structural properties are preserved by reductions. We illustrate this using example of chains in Lemma 5.3 below.

We first define chains as follows.

**Definition 5.1.** [19] A *chain* is a sequence of languages  $L_1, L_2, \dots, L_j$ , such that  $L_1 \subset L_2 \subset \dots \subset L_j$ .

If  $L_1, L_2, \dots, L_j$  form a chain, then we also refer to them as a *j-chain*.

We say that two chains  $L_1, L_2, \dots, L_j$  and  $L'_1, L'_2, \dots, L'_k$  are independent iff they do not contain any language in common.

We say that  $\mathcal{L}$  contains a *j-chain*, iff it contains languages  $L_1, L_2, \dots, L_j$  which form a *j-chain*. Similarly, we say that  $\mathcal{L}$  contains *k-independent j-chains* iff, for  $1 \leq r \leq k, 1 \leq i \leq j$ ,  $\mathcal{L}$  contains languages  $L_i^r$ , such that, for  $1 \leq r \leq k, L_1^r, L_2^r, \dots, L_j^r$  form pairwise-independent chains.

The next lemma gives sufficient condition for nonreducibility in the strong sense. It says for  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ ,  $\mathcal{L}'$  must contain at least as many pairwise-independent *j-chains*, as  $\mathcal{L}$ .

**Lemma 5.3.** [19] Let  $j > 0$ . Suppose  $\mathcal{L}$  contains *k* pairwise-independent *j-chains* and  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ . Then  $\mathcal{L}'$  also has *k* pairwise-independent *j-chains*.

A slightly weaker version of the above lemma holds for weak reduction.

**Lemma 5.4.** [19] Suppose  $\mathcal{L}, \mathcal{L}' \subseteq \mathcal{E}$ . Suppose  $\mathcal{L} \leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}'$ . Suppose further that  $\mathcal{L}$  contains *k* pairwise-independent *j-chains*, and  $\mathcal{L}'$  consists only of finite languages. Then,  $\mathcal{L}'$  contains *k* pairwise-independent *j-chains*.

One can generalize above lemmas to show that several structural constraints must be satisfied in any reduction. This often allows us to claim nonreducibility results just based on structure.

## 6. Intrinsic Complexity of Natural Classes

In this section we will present some results about the intrinsic complexity relationship among several natural classes.

Let us first define some natural classes.

**Definition 6.1.**  $\text{SINGLE} = \{L \mid \text{card}(L) = 1\}$ .  
 $\text{COSINGLE} = \{L \mid \text{card}(N - L) = 1\}$ .  
 $\text{INIT} = \{L \mid (\exists i \in N)[L = \{x \mid x \leq i\}]\}$ .  
 $\text{COINIT} = \{L \mid (\exists i \in N)[L = \{x \mid x \geq i\}]\}$ .  
 $\text{FIN} = \{L \mid \text{card}(L) < \infty\}$ .

The following theorem shows that SINGLE is strictly simpler learning problem than COINIT with respect to both  $\leq_{\text{strong}}^{\text{TxtEx}}$  and  $\leq_{\text{weak}}^{\text{TxtEx}}$  reductions.

**Theorem 6.1.** [17]  $\text{SINGLE} \leq_{\text{strong}}^{\text{TxtEx}} \text{COINIT} \wedge \text{COINIT} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{SINGLE}$ .

The following result justifies the earlier discussion that COINIT is a simpler learning problem than FIN.

**Theorem 6.2.** [17]  $\text{COINIT} \leq_{\text{weak}}^{\text{TxtEx}} \text{FIN} \wedge \text{FIN} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{COINIT}$ .

**Proof:**

$\text{COINIT} \leq_{\text{weak}}^{\text{TxtEx}} \text{FIN}$  follows from Theorem 7.1 presented later.  $\text{FIN} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{COINIT}$  follows from Theorem 9.6 presented later.  $\square$

In contrast to above result the following theorem shows that  $\text{COINIT} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ .

**Theorem 6.3.** [17]  $\text{COINIT} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ .

**Proof:**

Suppose by way of contradiction that  $\text{COINIT} \leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ , as witnessed by  $\Theta$  and  $\Psi$ . Then by Corollary 5.1 it follows that  $(\forall L \in \text{COINIT})[\Theta(L) \subseteq \Theta(N)]$ . Since COINIT is an infinite collection of languages, it follows that either  $\Theta(N)$  is infinite or there exist distinct  $L_1$  and  $L_2$  in COINIT such that  $\Theta(L_1) = \Theta(L_2)$ . It follows that  $\text{COINIT} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ .  $\square$

Our next result shows that INIT and FIN are equivalent in the strong sense.

**Theorem 6.4.** [17]  $\text{INIT} \equiv_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ .

Next three results establish the position of COSINGLE among the classes SINGLE, INIT, COINIT, FIN.

**Theorem 6.5.** [17]  $\text{SINGLE} \leq_{\text{strong}}^{\text{TxtEx}} \text{COSINGLE}$ .  $\text{COSINGLE} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{SINGLE}$ .

**Theorem 6.6.** [17]  $\text{COINIT} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{COSINGLE}$ .

$\text{COINIT} \leq_{\text{weak}}^{\text{TxtEx}} \text{COSINGLE}$ .

$\text{COSINGLE} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{COINIT}$ .

**Theorem 6.7.** [17]  $\text{COSINGLE} \leq_{\text{strong}}^{\text{TxtEx}} \text{INIT}$ .

$\text{INIT} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{COSINGLE}$ .

$\text{INIT} \equiv_{\text{weak}}^{\text{TxtEx}} \text{COSINGLE}$ .

Earlier results about identification in the limit from positive data turned out to be pessimistic because Gold [11] established that any collection of languages that contains an infinite language and all its finite subsets cannot be  $\text{TxtEx}$ -identified. As a consequence of this result no class in the Chomsky hierarchy can be identified in the limit from texts. However, later, two interesting classes were proposed that could be identified in the limit from texts. In this section, we describe these classes and locate their status with respect to the reductions introduced in this paper.

The first of these classes was introduced by Wiehagen [29]. We define,  $\text{WIEHAGEN} = \{L \mid L \in \mathcal{E} \wedge L = W_{\min(L)}\}$ .

$\text{WIEHAGEN}$  is an interesting class because it can be shown that it contains a finite variant of every recursively enumerable language. It is easy to verify that  $\text{WIEHAGEN} \in \text{TxtEx}$ . It is also easy to see that there exists a machine which  $\text{TxtEx}$ -identifies  $\text{WIEHAGEN}$  and that this machine, while processing a text for any language in  $\text{WIEHAGEN}$ , can provide an upper bound on the number of additional mind changes required before convergence. In this connection this class appears to pose a learning problem similar in nature to  $\text{COINIT}$  above. This intuition is indeed justified by the following theorem as these two classes turn out to be equivalent in the strong sense.

**Theorem 6.8.** [17]  $\text{COINIT} \equiv_{\text{strong}}^{\text{TxtEx}} \text{WIEHAGEN}$ .

We next consider the class,  $\text{PATTERN}$ , of pattern languages introduced by Angluin [1].

Suppose  $V$  is a countably infinite set of variables and  $C$  is a nonempty *finite* set of constants, such that  $V \cap C = \emptyset$ . *Notation:* For a set  $X$  over variables and constants,  $X^*$  denotes the set of strings over  $X$ , and  $X^+$  denotes the set of nonempty strings over  $X$ . Any  $w \in (V \cup C)^+$  is called a *pattern*. Suppose  $f$  is a mapping from  $(V \cup C)^+$  to  $C^+$ , such that, for all  $a \in C$ ,  $f(a) = a$  and, for each  $w_1, w_2 \in (V \cup C)^+$ ,  $f(w_1 \cdot w_2) = f(w_1) \cdot f(w_2)$ , where  $\cdot$  denotes concatenation of strings. Let  $\text{PatMap}$  denote the collection of all such mappings  $f$ .

Let  $\text{code}$  denote a 1-1 onto mapping from strings in  $C^*$  to  $N$ .

The language associated with the pattern  $w$  is defined as  $\mathbf{Lang}(w) = \{\text{code}(f(w)) \mid f \in \text{PatMap}\}$ . Then,  $\text{PATTERN} = \{\mathbf{Lang}(w) \mid w \text{ is a pattern}\}$ .

The following theorem shows that learning  $\text{PATTERN}$  has the same complexity as  $\text{COINIT}$  and  $\text{WIEHAGEN}$ .

**Theorem 6.9.** [17]  $\text{COINIT} \equiv_{\text{strong}}^{\text{TxtEx}} \text{PATTERN}$ .

**Proof:**

We first show that  $\text{COINIT} \leq_{\text{strong}}^{\text{TxtEx}} \text{PATTERN}$ . Let  $S_i = \mathbf{Lang}(a^i x)$ , where  $a \in C$  and  $x \in V$ . Let  $\Theta$  be such that  $\Theta(L) = S_{\min(L)} = \{\text{code}(a^l w) \mid w \in C^+ \wedge l = \min(L)\}$ . Note that such a  $\Theta$  can be easily constructed. Note that  $\text{code}(a^{l+1}) \in \text{content}(\Theta(L)) \Leftrightarrow l \geq \min(L)$ .

Let  $f(i)$  denote an index of a grammar (obtained effectively from  $i$ ) for  $\{x \mid x \geq i\}$ . Let  $\Psi$  be defined as follows. Suppose  $G = g_0, g_1, \dots$ . Then  $\Psi(G) = g'_0, g'_1, \dots$ , such that, for  $n \in N$ ,  $g'_n = f(\min(\{l \mid \text{code}(a^{l+1}) \in W_{g_n, n}\}))$ . It is easy to verify that  $\Theta$  and  $\Psi$  witness that  $\text{COINIT} \leq_{\text{strong TxE}} \text{PATTERN}$ .

We now show that  $\text{PATTERN} \leq_{\text{strong TxE}} \text{COINIT}$ . Note that there exists a recursive indexing  $L_0, L_1, \dots$  of pattern languages such that

- (1)  $L_i = L_j \Leftrightarrow i = j$ .
- (2)  $L_i \subset L_j \Rightarrow i > j$ .

(One such indexing can be obtained as follows. First note that for patterns  $w_1$  and  $w_2$ , if  $\mathbf{Lang}(w_1) \subseteq \mathbf{Lang}(w_2)$  then length of  $w_1$  is at least as large as that of  $w_2$ . Also for patterns of the same length  $\subseteq$  relation is decidable, as shown by Angluin [1]. Thus we can form the indexing as required using the following method. We consider only canonical patterns [1]. For  $w_1 \neq w_2$ , we place  $w_1$  before  $w_2$  if (a) length of  $w_1$  is smaller than that of  $w_2$  or (b) length of  $w_1$  and  $w_2$  are same, but  $\mathbf{Lang}(w_1) \supseteq \mathbf{Lang}(w_2)$  or (c) length of  $w_1$  and  $w_2$  are same,  $\mathbf{Lang}(w_1) \not\subseteq \mathbf{Lang}(w_2)$  and  $w_1$  is lexicographically smaller than  $w_2$ .)

Moreover, there exists a machine,  $\mathbf{M}$ , such that

- (a) For all  $\sigma \subseteq \tau$ , such that  $\text{content}(\sigma) \neq \emptyset$ ,  $\mathbf{M}(\sigma) \geq \mathbf{M}(\tau)$ .
- (b) For all texts  $T$  for pattern languages,  $\mathbf{M}(T) \downarrow = i$ , such that  $L_i = \text{content}(T)$ .

(Angluin's method of identification of pattern languages essentially achieves this property).

Let  $\tau_{m,n}$  be the lexicographically least sequence of length  $n$ , such that  $\text{content}(\tau_{m,n}) = \{x \mid m \leq x \leq n\}$ . Let  $\text{prev}(\Lambda) = \Lambda$ ; for  $w \in \{N\} \cup \{\#\}$  and  $\sigma = \sigma' \diamond w$ , let  $\text{prev}(\sigma)$  denote  $\sigma'$ .

If  $\text{content}(\sigma) = \emptyset$ , then  $\Theta(\sigma) = \sigma$ , else  $\Theta(\sigma) = \Theta(\text{prev}(\sigma)) \diamond \tau_{\mathbf{M}(\sigma), |\sigma|}$ . Note that for a text  $T$  for  $L_i$ ,  $\Theta(T)$  would be  $\{x \mid x \geq \mathbf{M}(T)\}$ .

Let  $f(i)$  denote a grammar effectively obtained from  $i$  for  $L_i$ . Let  $\Psi$  be defined as follows. Suppose  $G = g_0, g_1, \dots$ . Then  $\Psi(G) = g'_0, g'_1, \dots$ , such that, for  $n \in N$ ,  $g'_n = f(\min(\{n\} \cup W_{g_n, n}))$ . It is easy to verify that if  $G$  converges to a grammar for  $\{x \mid x \geq i\}$ , then  $\Psi$  converges to a grammar for  $L_i$ . Thus,  $\Theta$  and  $\Psi$  witness that  $\text{PATTERN} \leq_{\text{strong TxE}} \text{COINIT}$ .  $\square$

There have been several other "natural" classes studied in the literature, for example  $\text{CONTON}_n = \{L \mid \text{card}(N - L) = n\}$ . We however will not consider them in this survey. Jain, Kinber and Wiehagen [14] considered generalization of the above natural classes to multidimensional languages where each individual dimension is learnable by using one of above strategies (such as INIT, COINIT, COSINGLE). Recently Jain and Kinber [13] have also considered intrinsic complexity of learning several natural geometrical classes such as semi-hulls and open-hulls.

## 7. Complete Classes

Complete classes have often been used to identify the hardest problem in a class of problems. In this section we will study some problems which are known to be complete for weak and strong reductions for  $\text{TxE}$  identification.

**Theorem 7.1.** [17] COSINGLE, INIT and FIN are  $\leq_{\text{weak TxE}}$ -complete.

**Proof:**

We only show FIN is  $\leq_{\text{weak TxE}}$ -complete. INIT and COSINGLE can be shown to be  $\leq_{\text{weak TxE}}$ -complete in a similar fashion.

Consider any  $\mathcal{L} \in \mathbf{TxtEx}$ . Suppose  $\mathcal{L} \subseteq \mathbf{TxtEx}(\mathbf{M})$ . We construct  $\Theta$  and  $\Psi$  which witness that  $\mathcal{L} \leq_{\text{weak}}^{\mathbf{TxtEx}} \text{FIN}$ . Without loss of generality we assume that  $\mathbf{M}(T[0]) = ?$ . Define  $\Theta(T[n])$  as follows.  
 $\Theta(T[0]) = \text{sequence containing just } \#$ .

$$\Theta(T[n+1]) = \begin{cases} \Theta(T[n]) \diamond \#, & \text{if } \mathbf{M}(T[n]) = \mathbf{M}(T[n+1]); \\ \Theta(T[n]) \diamond \langle \mathbf{M}(T[n+1]), n+1 \rangle, & \text{otherwise.} \end{cases}$$

If  $\mathbf{M}$   $\mathbf{TxtEx}$ -identifies  $T$ , then  $\text{content}(\Theta(T))$  is a nonempty finite set. Let  $j_T = \max(\{n \mid (\exists i)[\langle i, n \rangle \in \text{content}(\Theta(T))]\})$ . One can now easily verify that, if  $\mathbf{M}$   $\mathbf{TxtEx}$ -identifies  $T$ , then  $\mathbf{M}(T)$  is the unique  $i$  such that  $\langle i, j_T \rangle \in \text{content}(\Theta(T))$ .

Suppose  $\alpha = g_0 g_1 g_2 \dots$  is an infinite sequence of grammars. Define  $\Psi(\alpha) = g'_0, g'_1, g'_2, \dots$ , where  $g'_m$  is defined as follows. Let  $j_m = \max(\{n \mid (\exists i)[\langle i, n \rangle \in W_{g_m, m}]\})$ . Let  $g'_m = \min(\{i \mid \langle i, j_m \rangle \in W_{g_m, m}\})$ .

It is easy to verify that, if  $\mathbf{M}$   $\mathbf{TxtEx}$ -identifies  $T$ , and  $\alpha$  converges to a grammar for  $\text{content}(\Theta(T))$ , then  $\Psi(\alpha)$  converges to  $\mathbf{M}(T)$ , a grammar for  $\text{content}(T)$ . It follows that  $\mathcal{L} \leq_{\text{weak}}^{\mathbf{TxtEx}} \text{FIN}$ .  $\square$

However, using Theorem 6.3, we have

**Corollary 7.1.** FIN, COSINGLE, INIT are not  $\leq_{\text{strong}}^{\mathbf{TxtEx}}$ -complete.

We next consider a natural class which is  $\leq_{\text{strong}}^{\mathbf{TxtEx}}$ -complete.

Let  $Q$  denote the set of all rational numbers  $\geq 0$ . For  $s, r \in Q$ , let  $Q_{s,r} = \{x \in Q \mid s \leq x \leq r\}$ . For allowing us to consider r.e. sets of rational numbers, let  $\text{coderat}(\cdot)$  denote an effective bijective mapping from  $Q$  to  $N$ .

**Definition 7.1.** Suppose  $r \in Q_{0,1}$ .

Let  $X_r = \{\text{coderat}(x) \mid x \in Q \text{ and } 0 \leq x \leq r\}$ .

**Definition 7.2.** Suppose  $s, r \in Q_{0,1}$  and  $s < r$ .

Let  $\text{RINIT}_{s,r} = \{X_w \mid w \in Q_{s,r}\}$ .

**Theorem 7.2.** [14]  $\text{RINIT}_{0,1}$  is  $\leq_{\text{strong}}^{\mathbf{TxtEx}}$ -complete.

## 8. Characterizations

Characterizations are often useful in theoretical studies and specially in inductive inference (for example see the survey by Zeugmann and Lange [32]). In this section we will be considering some characterizations.

We first consider characterization of complete classes. For this we introduce the notion of limiting-standardizable.

**Definition 8.1.** [20, 9, 16] A class  $\mathcal{L}$  of recursively enumerable sets is called *limiting standardizable* iff there exists a partial limiting recursive function  $F$  such that

(a) For all  $i$  such that  $W_i = L$  for some  $L \in \mathcal{L}$ ,  $F(i)$  is defined.

(b) For all  $L, L' \in \mathcal{L}$ , for all  $i, j$  such that  $W_i = L$  and  $W_j = L'$ ,

$$F(i) = F(j) \Leftrightarrow L = L'.$$

Thus, informally, a class  $\mathcal{L}$  of r.e. languages is limiting standardizable if all the infinitely many grammars  $i \in N$  of each language  $L \in \mathcal{L}$  can be mapped (“standardized”) in the limit to some unique grammar (natural number). Notice that it is not required that this “standard grammar” must be a grammar of  $L$  again. However, standard grammars for *different* languages from  $\mathcal{L}$  have to be pairwise different.

The following theorem characterizes the  $\leq_{\text{strong}}^{\text{TxtEx}}$ -complete classes.

**Theorem 8.1.** [14] Suppose  $\mathcal{L} \in \text{TxtEx}$ . Then, the following three statements are equivalent.

- (1)  $\mathcal{L}$  is  $\leq_{\text{strong}}^{\text{TxtEx}}$ -complete.
- (2)  $\text{RINIT}_{0,1} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}$ .
- (3) There exists a recursive function  $H$  from  $Q_{0,1}$  to  $N$  such that:

- (a)  $\{W_{H(r)} \mid r \in Q_{0,1}\} \subseteq \mathcal{L}$ .
- (b) If  $0 \leq r < r' \leq 1$ , then  $W_{H(r)} \subset W_{H(r')}$ .
- (c)  $\{W_{H(r)} \mid r \in Q_{0,1}\}$  is limiting standardizable.

Intuitively  $H$  in part (c) in the above characterization gives a subclass of  $\mathcal{L}$ , which is in some sense effectively isomorphic to  $\text{RINIT}_{0,1}$ .

We next give a characterization of  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete classes.

**Definition 8.2.** [14] A nonempty class  $\mathcal{L}$  of languages is called *quasi-dense* iff

- (a)  $\mathcal{L}$  is 1–1 recursively enumerable.
- (b) For any  $L \in \mathcal{L}$  and any finite  $S \subseteq L$ , there exists an  $L' \in \mathcal{L}$ , such that  $S \subseteq L'$ , but  $L \neq L'$ .

Note: (b) can be equivalently replaced by

(b') For any finite set  $S$ , either there exists no language in  $\mathcal{L}$  extending  $S$ , or there exist infinitely many distinct languages in  $\mathcal{L}$  extending  $S$ .

**Theorem 8.2.** [14]  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete iff  $\mathcal{L} \in \text{TxtEx}$  and there exists a quasi-dense subclass of  $\mathcal{L}$  which is limiting standardizable.

Jain, Kinber and Wiehagen [14] also give several other characterizations of classes which are reducible to natural classes such as INIT, and classes to which natural classes are reducible. For example, strong-degrees below and above INIT can be characterized as follows:

**Definition 8.3.** [14]  $F$ , a partial recursive mapping from  $\text{FIN} \times N$  to  $N$ , is called an *up-mapping* iff for all finite sets  $S, S'$ , for all  $j, j' \in N$ :

$$\text{If } S \subseteq S' \text{ and } j \leq j', \text{ then } F(S, j) \downarrow \Rightarrow [F(S', j') \downarrow \geq F(S, j)].$$

For an up-mapping  $F$  and  $L \subseteq N$ , we abuse notation slightly and let  $F(L)$  denote  $\lim_{S \rightarrow L, j \rightarrow \infty} F(S, j)$  (where by  $S \rightarrow L$  we mean: take any sequence of finite sets  $S_1, S_2, \dots$ , such that  $S_i \subseteq S_{i+1}$  and  $\bigcup S_i = L$ , and then take the limit over these  $S_i$ 's).

Note that  $F(L)$  may be undefined in two ways:

- (1)  $F(S, j)$  may take arbitrarily large values for  $S \subseteq L$ , and  $j \in N$ , or
- (2)  $F(S, j)$  may be undefined for all  $S \subseteq L$ ,  $j \in N$ .

**Theorem 8.3.** [14]  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \text{INIT}$  iff there exist  $F$ , a partial recursive up-mapping, and  $G$ , a partial limit recursive mapping from  $N$  to  $N$ , such that

- (a) For any language  $L \in \mathcal{L}$ ,  $F(L) \downarrow < \infty$ .
- (b) For all  $L \in \mathcal{L}$ ,  $G(F(L))$  converges to a grammar for  $L$ .

**Theorem 8.4.** [14]  $\text{INIT} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}$  iff there exists a recursive function  $H$  such that

- (a)  $\{W_{H(i)} \mid i \in N\} \subseteq \mathcal{L}$ ,
- (b)  $W_{H(i)} \subset W_{H(i+1)}$ , and
- (c)  $\{W_{H(i)} \mid i \in N\}$  is limiting standardizable.

Several similar characterization for natural classes such as COINIT and COSINGLE were also given by Jain, Kinber and Wiehagen [14]. The above paper also considers a generalization of classes such as INIT and COINIT, by considering combinations of INIT and COINIT-like strategies. They give a hierarchy based on such combinations, and characterize the learning classes so formed.

## 9. Some structural Results

We now consider some structural results regarding reductions. The next theorem shows that there exists an infinite hierarchy of more and more complex classes.

Let  $\text{FIN}_i = \{L \mid \text{card}(L) \leq i\}$ .

**Theorem 9.1.** [19] For each  $i \geq 1$ ,  $\text{FIN}_i \leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}_{i+1}$  and  $\text{FIN}_{i+1} \not\leq_{\text{weak}}^{\text{TxtEx}} \text{FIN}_i$ .

One can view the reducibility structure as a directed graph, where nodes represent language classes, and an edge from  $\mathcal{L}$  to  $\mathcal{L}'$  denotes the fact that  $\mathcal{L}$  is (weak, strong) reducible to  $\mathcal{L}'$ .

Theorem 9.2 shows that the structure of intrinsic complexity is very rich as any finite acyclic directed graph can be embedded in this structure.

**Theorem 9.2.** [19] Every finite directed acyclic graph  $H$  can be embedded in the reducibility structure.

Ambainis (private communication) has shown that any recursively enumerable DAG (even infinite) can be embedded in the reducibility structure.

Although the above theorem shows that the intrinsic complexity of language identification is rich, the next two results establish that this structure is not *dense*, that is, there exist two language classes,  $\mathcal{L}$  and  $\mathcal{L}'$ , that satisfy the following properties:

- (a)  $\mathcal{L}$  is strong-reducible to  $\mathcal{L}'$  but  $\mathcal{L}'$  is not even weak-reducible to  $\mathcal{L}$ .
- (b) There is no language class between  $\mathcal{L}$  and  $\mathcal{L}'$  with respect to either strong or weak reduction.

**Theorem 9.3.** [19] For  $i > 0$ , let  $L_i = \{i\}$ . Let  $L_0 = \{1, 0\}$ . Let  $\mathcal{L} = \{L_i \mid i > 0\}$ . Let  $\mathcal{L}' = \{L_0\} \cup \mathcal{L}$ . (Note that  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ , but  $\mathcal{L}' \not\leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}$ ). Then for all  $\mathcal{S}$  such that  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{S} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ , either  $\mathcal{S} \equiv_{\text{strong}}^{\text{TxtEx}} \mathcal{L}$  or  $\mathcal{S} \equiv_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ .

**Theorem 9.4.** [19] For  $i > 0$ , let  $L_i = \{i\}$ . Let  $L_0 = \{1, 0\}$ . Let  $\mathcal{L} = \{L_i \mid i > 0\}$ . Let  $\mathcal{L}' = \{L_0\} \cup \mathcal{L}$ . (Note that  $\mathcal{L} \leq_{\text{strong}}^{\text{TxtEx}} \mathcal{L}'$ , but  $\mathcal{L}' \not\leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}$ ). Then for all  $\mathcal{S}$  such that  $\mathcal{L} \leq_{\text{weak}}^{\text{TxtEx}} \mathcal{S} \leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}'$ , either  $\mathcal{S} \equiv_{\text{weak}}^{\text{TxtEx}} \mathcal{L}$  or  $\mathcal{S} \equiv_{\text{weak}}^{\text{TxtEx}} \mathcal{L}'$ .

We have seen earlier that FIN is complete with respect to  $\leq_{\text{weak}}^{\text{TxtEx}}$  reduction. This means that FIN captures the essence of the most difficult learning problem with respect to weak-reduction. It was also shown that FIN is not complete with respect to strong-reduction. Below we give an interesting collection of languages that is trivially identifiable (with 0 mind changes) but is not strong reducible to FIN.

**Theorem 9.5.** [19] Let  $\mathcal{L} = \{L \mid L \neq \emptyset \wedge (\forall x \in L)[W_x = L]\}$ . Then  $\mathcal{L} \not\leq_{\text{strong}}^{\text{TxtEx}} \text{FIN}$ .

We now consider connections between topological properties of learnable classes and their intrinsic complexity. The following notion was introduced by Angluin [1].

**Definition 9.1.** [1]  $\mathcal{L}$  has *finite thickness* just in case for each  $n \in N$ ,  $\text{card}(\{L \in \mathcal{L} \mid n \in L\})$  is finite.

PATTERN has finite thickness. Angluin [1] showed that if  $\mathcal{L}$  is an indexed family of recursive languages and  $\mathcal{L}$  has finite thickness then  $\mathcal{L} \in \text{TxtEx}$ .

We now present a theorem that turns out to be very useful in showing that certain classes are not complete with respect to  $\leq_{\text{weak}}^{\text{TxtEx}}$  reduction. The theorem states that if a collection of languages  $\mathcal{L}$  is such that each natural number  $x$  appears in only finitely many languages in  $\mathcal{L}$ , then FIN is not  $\leq_{\text{weak}}^{\text{TxtEx}}$  reducible to  $\mathcal{L}$ . Since  $\text{FIN} \in \text{TxtEx}$ , this theorem immediately implies that classes such as COINIT, PATTERN, WIEHAGEN are not  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete.

**Theorem 9.6.** [17] Suppose  $\mathcal{L}$  has finite thickness. Then  $\text{FIN} \not\leq_{\text{weak}}^{\text{TxtEx}} \mathcal{L}$ .

A more interesting topological notion was introduced by Wright [31] described below (see also paper by Motoki, Shinohara, and Wright [26]).

**Definition 9.2.** [31, 26]  $\mathcal{L}$  has *infinite elasticity* just in case there exists an infinite sequence of pairwise distinct numbers,  $\{w_i \in N \mid i \in N\}$ , and an infinite sequence of pairwise distinct languages,  $\{A_i \in \mathcal{L} \mid i \in N\}$ , such that for each  $k \in N$ ,  $\{w_i \mid i < k\} \subseteq A_k$ , but  $w_k \notin A_k$ .  $\mathcal{L}$  is said to have *finite elasticity* just in case  $\mathcal{L}$  does not have infinite elasticity.

Wright [31] showed that if a class  $\mathcal{L}$  has finite thickness then it has finite elasticity. He further showed that if a class  $\mathcal{L}$  is an indexed family of recursive languages and  $\mathcal{L}$  has finite elasticity, then  $\mathcal{L} \in \text{TxtEx}$ .

Now, language classes that are  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete are, in some sense, the most difficult learning problems. Interestingly, it has been established that  $\leq_{\text{weak}}^{\text{TxtEx}}$ -completeness is also a sufficient condition for infinite elasticity.

**Theorem 9.7.** [18] Suppose  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete. Then  $\mathcal{L}$  has infinite elasticity.

Classes that have infinite elasticity are not necessarily identifiable. However, it is interesting to ask: Are all identifiable classes that have infinite elasticity also  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete? The following result answers this question negatively.

**Theorem 9.8.** [18] There exist a class  $\mathcal{L}$  such that  $\mathcal{L} \in \text{TxtEx}$  and  $\mathcal{L}$  has infinite elasticity, but  $\mathcal{L}$  is not  $\leq_{\text{weak}}^{\text{TxtEx}}$ -complete.

## 10. Informants

In this section we briefly consider intrinsic complexity of learning from informants. The concepts of weak and strong reduction can be adopted to language identification from informants. Informally, informants, first introduced by Gold [11], are texts which contain both positive and negative data. Thus if  $I_L$  is an informant for  $L$ , then  $\text{content}(I_L) = \{\langle x, 0 \rangle \mid x \notin L\} \cup \{\langle x, 1 \rangle \mid x \in L\}$ .<sup>1</sup> Identification in the limit from informants is referred to as **InfEx**-identification (we refer the reader to [11] for details). The definition of weak and strong reduction can be adopted to language identification from informants in a straightforward way by replacing texts by informants in Definitions 4.1 and 4.3.

For any language  $L$ , an informant of special interest is the canonical informant.  $I$  is a *canonical* informant for  $L$  just in case for  $n \in N$ ,  $I(n) = \langle n, \chi_L(n) \rangle$ , where  $\chi_L$  denotes the characteristic function of  $L$ .

Since a canonical informant can always be produced from any informant, we have the following:

**Proposition 10.1.**  $\mathcal{L}_1 \leq_{\text{weak}}^{\text{InfEx}} \mathcal{L}_2 \iff \mathcal{L}_1 \leq_{\text{strong}}^{\text{InfEx}} \mathcal{L}_2$ .

**Theorem 10.1.** [17] FIN is  $\leq_{\text{strong}}^{\text{InfEx}}$  complete.

However,

**Theorem 10.2.** [17] The classes SINGLE, INIT, COSINGLE, COINIT, WIEHAGEN, and PATTERN are equivalent with respect to  $\leq_{\text{strong}}^{\text{InfEx}}$  reduction.<sup>2</sup>

## 11. Function Learning

In this section we briefly consider intrinsic complexity of function learning. Freivalds, Kinber and Smith [10] were the first to consider intrinsic complexity of function learning. In this survey we mostly follow Kinber, Papazian, Smith, and Wiehagen [22].

We first consider some notation and definitions for function identification.

For a function  $\eta$  such that  $\eta(x) \downarrow$ , for  $x < n$ , we let  $\eta[n] = \{(x, \eta(x)) \mid x < n\}$ . We let  $\text{SEG} = \{f[n] \mid f \in \mathcal{R}\}$ . A function learning machine is an algorithmic mapping from  $\text{SEG}$  into  $N$ .

**Definition 11.1.** [11, 6]

- (a) **M Ex**-identifies a function  $f$  (written  $f \in \mathbf{Ex}(\mathbf{M})$ ) just in case  $(\exists i \mid \varphi_i = f) (\forall n) [\mathbf{M}(f[n]) = i]$ .
- (b) **M Ex**-identifies a class  $\mathcal{C}$  of recursive functions (written:  $\mathcal{C} \subseteq \mathbf{Ex}(\mathbf{M})$ ) just in case **M Ex**-identifies each function from  $\mathcal{C}$ .
- (d)  $\mathbf{Ex} = \{\mathcal{C} \subseteq \mathcal{R} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Ex}(\mathbf{M})]\}$ .

In considering intrinsic complexity of function learning it is easier to consider  $\Theta$  as mapping partial functions to partial functions.

<sup>1</sup>Alternatively, an informant for a language  $L$  may be thought of as a “tagged” text for  $N$  such that  $n$  appears in the text with tag 1 if  $n \in L$ ; otherwise  $n$  appears in the text with tag 0.

<sup>2</sup>Actually, it can be shown that any collection of languages that can be finitely identified (i.e., identified with 0 mind changes) from informants is  $\leq_{\text{strong}}^{\text{InfEx}}$  SINGLE.

**Definition 11.2.** [28] A recursive operator is an effective total mapping  $\Theta$  from (possibly partial) functions to (possibly partial) functions, which satisfies the following properties:

- (a) **Monotonicity:** For all functions  $\eta$  and  $\eta'$ , if  $\eta \subseteq \eta'$ , then  $\Theta(\eta) \subseteq \Theta(\eta')$ .
- (b) **Compactness:** For all  $\eta$ , if  $(x, y) \in \Theta(\eta)$ , then there exists a finite function  $\alpha \subseteq \eta$  such that  $(x, y) \in \Theta(\alpha)$ .
- (c) **Recursiveness:** For all finite functions  $\alpha$ , one can effectively enumerate (in  $\alpha$ ) all  $(x, y) \in \Theta(\alpha)$ .

Admissible sequences for function learning criteria can be defined similarly to the language learning case.

**Definition 11.3.** [10, 22]. Suppose  $\mathcal{C}_1 \subseteq \mathcal{R}$ ,  $\mathcal{C}_2 \subseteq \mathcal{R}$ , and identification criteria  $\mathbf{I}_1$  and  $\mathbf{I}_2$  are given.

We say that  $\mathcal{C}_1 \leq^{\mathbf{I}_1, \mathbf{I}_2} \mathcal{C}_2$  iff there exist recursive operators  $\Theta$  and  $\Psi$  such that for any function  $f \in \mathcal{C}_1$ ,

1.  $\Theta(f) \in \mathcal{C}_2$ ,
2. for any  $\mathbf{I}_2$  admissible sequence  $\alpha$  for  $\Theta(f)$ ,  $\Psi(\alpha)$  is an  $\mathbf{I}_1$  admissible sequence for  $f$ .

We say that  $\mathcal{C}_1 \leq^{\mathbf{I}} \mathcal{C}_2$ , iff  $\mathcal{C}_1 \leq^{\mathbf{I}, \mathbf{I}} \mathcal{C}_2$ .  $\leq^{\mathbf{I}}$ -hardness and  $\leq^{\mathbf{I}}$ -completeness can be defined similarly.

Notice that unlike in the language learning case we haven't defined weak and strong reductions for function learning. Reason is that for most natural identification criteria for function learning, these two reductions are same. (Since one can effectively convert an arbitrary ordering of a total function into canonical order). Thus we only concentrate on strong reductions.

We next consider a complete class for **Ex**-identification.

Let  $\text{FINSUP} = \{f \in \mathcal{R} \mid \text{card}(\{x \mid f(x) \neq 0\}) < \infty\}$ .

**Theorem 11.1.** [10, 22]  $\text{FINSUP}$  is  $\leq^{\text{Ex}}$ -complete.

We now give characterizations for  $\leq^{\text{Ex}}$ -complete classes.

**Definition 11.4.** [28] A nonempty class  $\mathcal{C} \subseteq \mathcal{R}$  is said to be *recursively enumerable* iff there exists a recursive function  $f$  such that  $\mathcal{C} = \{\varphi_{f(i)} \mid i \in \mathbb{N}\}$ .

**Definition 11.5.** A function  $f$  is said to be an *accumulation point* of  $\mathcal{C}$ , iff for all  $n$ , there exists a  $g \in \mathcal{C}$  such that  $f(x) = g(x)$ , for  $x \leq n$ , but  $f \neq g$ .

**Definition 11.6.** [22]  $\mathcal{C}$  is called *dense* iff  $\mathcal{C}$  is nonempty and every  $f \in \mathcal{C}$  is an accumulation point of  $\mathcal{C}$ .

**Theorem 11.2.** [22]  $\mathcal{C}$  is  $\leq^{\text{Ex}}$ -complete iff  $\mathcal{C} \in \mathbf{Ex}$  and  $\mathcal{C}$  contains an r.e. dense subclass.

Kinber, Papazian, Smith, and Wiehagen [22] also give several results regarding identification criteria involving mind changes. Freivalds, Kinber and Smith [10] also consider various other formulations of reductions (such as space bounded reduction). We refer the reader to above two papers for further results on intrinsic complexity of function identification.

## 12. Learning with Anomalies

In this section we briefly discuss some of the results in intrinsic complexity in presence of anomalies in the final program conjectured by the learner.

Let us first consider some notation.  $L_1 \Delta L_2$  denotes the symmetric difference of  $L_1$  and  $L_2$ , that is  $L_1 \Delta L_2 = (L_1 - L_2) \cup (L_2 - L_1)$ . For a natural number  $a$ , we say that  $L_1 =^a L_2$ , iff  $\text{card}(L_1 \Delta L_2) \leq a$ . We say that  $L_1 =^* L_2$ , iff  $\text{card}(L_1 \Delta L_2) < \infty$ . Thus, we take  $n < * < \infty$ , for all  $n \in N$ . If  $L_1 =^a L_2$ , then we say that  $L_1$  is an  $a$ -variant of  $L_2$ .

We now define identification with anomalies.

**Definition 12.1.** [11, 6, 5] Suppose  $a \in N \cup \{*\}$ .

- (a)  $\mathbf{M\ TxtEx}^a$ -identifies a text  $T$  just in case  $(\exists i \mid W_i =^a \text{content}(T)) (\forall n) [\mathbf{M}(T[n]) = i]$ .
- (b)  $\mathbf{M\ TxtEx}^a$ -identifies an r.e. language  $L$  (written:  $L \in \mathbf{TxtEx}^a(\mathbf{M})$ ) just in case  $\mathbf{M\ TxtEx}^a$ -identifies each text for  $L$ .
- (c)  $\mathbf{M\ TxtEx}^a$ -identifies a class  $\mathcal{L}$  of r.e. languages (written:  $\mathcal{L} \subseteq \mathbf{TxtEx}^a(\mathbf{M})$ ) just in case  $\mathbf{M\ TxtEx}^a$ -identifies each language from  $\mathcal{L}$ .
- (d)  $\mathbf{TxtEx}^a = \{\mathcal{L} \subseteq \mathcal{E} \mid (\exists \mathbf{M}) [\mathcal{L} \subseteq \mathbf{TxtEx}^a(\mathbf{M})]\}$ .

Note that  $\mathbf{TxtEx}^0 = \mathbf{TxtEx}$ . Note that Definition of reduction and completeness we used in Definition 4.1, Definition 4.3, and Definition 4.2 are general and thus can be used for  $\mathbf{TxtEx}^a$ -identification too.

We next consider complete classes for  $\mathbf{TxtEx}^a$ -identification, and their characterization.

**Definition 12.2.** Suppose  $r \in Q_{0,1}$ .

$$\text{Let } X_r^{\text{cyl}} = \{\text{coderat}(2w + x) \mid x \in Q, w \in N \text{ and } 0 \leq x \leq r\}.$$

**Definition 12.3.** Suppose  $s, r \in Q_{0,1}$  and  $s < r$ .

$$\text{Let } \text{RINIT}_{s,r}^{\text{cyl}} = \{X_w^{\text{cyl}} \mid w \in Q_{s,r}\}.$$

**Theorem 12.1.** [14] For all  $a \in N$ ,  $\text{RINIT}_{0,1}^{\text{cyl}}$  is  $\leq_{\text{strong}}^{\mathbf{TxtEx}^a}$ -complete.

The following definition is a generalization of the definition of limiting standardizability considered by Kinber [20], Freivalds [9] and Jain and Sharma [16].

**Definition 12.4.** [14] Let  $a \in N \cup \{*\}$ . A class  $\mathcal{L}$  of recursively enumerable sets is called  $a$ -limiting standardizable iff there exists a partial limiting recursive function  $F$  such that

- (a) For all  $i$  such that  $W_i =^a L$  for some  $L \in \mathcal{L}$ ,  $F(i)$  is defined.
- (b) For all  $L, L' \in \mathcal{L}$ , for all  $i, j$  such that  $W_i =^a L$  and  $W_j =^a L'$ ,

$$F(i) = F(j) \Leftrightarrow L = L'.$$

The following theorem characterizes the  $\leq_{\text{strong}}^{\mathbf{TxtEx}^a}$ -complete classes, for all  $a \in N$ .

**Theorem 12.2.** [14] Suppose  $a \in N$ . Suppose  $\mathcal{L} \in \mathbf{TxtEx}^a$ . Then, the following three statements are equivalent.

- (1)  $\mathcal{L}$  is  $\leq_{\text{strong}}^{\mathbf{TxtEx}^a}$ -complete.
- (2)  $\text{RINIT}_{0,1}^{\text{cyl}} \leq_{\text{strong}}^{\mathbf{TxtEx}^a} \mathcal{L}$ .
- (3) There exists a recursive function  $H$  from  $Q_{0,1}$  to  $N$  such that:
  - (a)  $\{W_{H(r)} \mid r \in Q_{0,1}\} \subseteq \mathcal{L}$ .
  - (b) If  $0 \leq r < r' \leq 1$ , then  $W_{H(r)} \subset W_{H(r')}$ .
  - (c)  $\{W_{H(r)} \mid r \in Q_{0,1}\}$  is  $a$ -limiting standardizable.

The following theorem characterizes the  $\leq_{\text{weak}}^{\mathbf{TxtEx}^a}$ -complete classes, for all  $a \in N \cup \{*\}$ .

**Theorem 12.3.** [14] Suppose  $a \in N \cup \{*\}$ .  $\mathcal{L}$  is  $\leq_{\text{weak}}^{\mathbf{TxtEx}^a}$ -complete iff  $\mathcal{L} \in \mathbf{TxtEx}^a$  and there exists a quasi-dense subclass of  $\mathcal{L}$  which is  $a$ -limiting standardizable.

For function learning, identification with anomalies can be defined similarly to language learning.

Notation:  $\eta_1 =^a \eta_2$ , iff  $\text{card}(\{x \mid \eta_1(x) \neq \eta_2(x)\}) \leq a$ .

**Definition 12.5.** [11, 6] Suppose  $a \in N \cup \{*\}$ .

- (a)  $\mathbf{MEx}^a$ -identifies a function  $f$  (written  $f \in \mathbf{Ex}^a(\mathbf{M})$ ) just in case  $(\exists i \mid \varphi_i =^a f)(\forall n)[\mathbf{M}(f[n]) = i]$ .
- (b)  $\mathbf{MEx}^a$ -identifies a class  $\mathcal{C}$  of recursive functions (written:  $\mathcal{C} \subseteq \mathbf{Ex}^a(\mathbf{M})$ ) just in case  $\mathbf{MEx}^a$ -identifies each function from  $\mathcal{C}$ .
- (d)  $\mathbf{Ex}^a = \{\mathcal{C} \subseteq \mathcal{R} \mid (\exists \mathbf{M})[\mathcal{C} \subseteq \mathbf{Ex}^a(\mathbf{M})]\}$ .

FINSUP serves as a complete class even for  $\mathbf{Ex}^a$ , for  $a \in N$ .

**Theorem 12.4.** [10, 22] Let  $a \in N$ . FINSUP is  $\leq^{\mathbf{Ex}^a}$ -complete.

For  $\mathbf{Ex}^*$ -identification complete classes take a slightly different form.

Kinber, Papazian, Smith, and Wiehagen [22] defined functions with quasi-finite support as follows.  $f$  has quasi-finite-support if

- (1) For all  $x \in N$ , if  $x$  is 0, 1 or not a power of prime, then  $f(x) = 0$ .
- (2) For all but finitely many prime numbers  $p$ , for all  $k \in N$ ,  $f(p^k) = 0$ .
- (3) For every prime number  $p$ , there are  $y$  and  $n \in N$  such that either  $f(p^k) = y$ , for all  $k \geq 1$ , OR

$$f(p^k) = \begin{cases} y, & \text{if } 1 \leq k \leq n; \\ 0, & \text{otherwise.} \end{cases}$$

Let  $\text{QUASIFINSUP} = \{f \mid f \text{ has quasi-finite-support}\}$ .

**Theorem 12.5.** [22]  $\text{QUASIFINSUP}$  is  $\leq^{\mathbf{Ex}^*}$ -complete.

Following two theorems give a characterization for  $\leq^{\mathbf{Ex}^a}$ -complete classes.

**Theorem 12.6.** [22] Suppose  $a \in N$ . Then  $\mathcal{C}$  is  $\leq^{\mathbf{Ex}^a}$ -complete iff  $\mathcal{C} \in \mathbf{Ex}^a$  and  $\mathcal{C}$  contains a r.e. dense subclass.

**Theorem 12.7.** [22]  $\mathcal{C}$  is  $\leq^{\text{Ex}^*}$ -complete iff  $\mathcal{C} \in \text{Ex}^*$  and  $\mathcal{C}$  contains a r.e. dense subclass  $\mathcal{S}$  such that for any two distinct  $f, g \in \mathcal{S}$ ,  $f \neq^* g$ .

### 13. Acknowledgements

We thank Arun Sharma for several helpful discussions and comments. Many of the results presented in this survey is joint work with Efim Kinber, Arun Sharma, Carl Smith and Rolf Wiehagen. The results discussed in this survey, and many of their extensions can be found in the papers [9, 10, 22, 13, 14, 17, 18, 19, 22]. Sanjay Jain was supported in part by NUS grant R252-000-127-112.

### References

- [1] Angluin, D.: Finding patterns common to a set of strings, *Journal of Computer and System Sciences*, **21**, 1980, 46–62.
- [2] Bārzdiņš, J., Freivalds, R.: On the prediction of General Recursive Functions, *Soviet Mathematics Doklady*, **13**, 1972, 1224–1228.
- [3] Blum, M.: A Machine-Independent Theory of the Complexity of recursive Functions, *Journal of the ACM*, **14**, 1967, 322–336.
- [4] Case, J.: The Power of Vacillation in Language Learning, *SIAM Journal on Computing*, **28**, 1999, 1941–1969.
- [5] Case, J., Lynes, C.: Machine Inductive Inference and Language Identification, *Proceedings of the 9th International Colloquium on Automata, Languages and Programming* (M. Nielsen, E. M. Schmidt, Eds.), 140, Springer-Verlag, 1982.
- [6] Case, J., Smith, C.: Comparison of Identification Criteria for Machine Inductive Inference, *Theoretical Computer Science*, **25**, 1983, 193–220.
- [7] Daley, R., Smith, C.: On the Complexity of Inductive Inference, *Information and Control*, **69**, 1986, 12–40.
- [8] Feldman, J.: Some Decidability results on Grammatical Inference and Complexity, *Information and Control*, **20**, 1972, 244–262.
- [9] Freivalds, R.: Inductive Inference of Recursive Functions: Qualitative Theory, in: *Baltic Computer Science* (J. Bārzdiņš, D. Bjorner, Eds.), vol. 502 of *Lecture Notes in Computer Science*, Springer-Verlag, 1991, 77–110.
- [10] Freivalds, R., Kinber, E., Smith, C.: On the Intrinsic Complexity of Learning, *Second European Conference on Computational Learning Theory* (P. Vitányi, Ed.), 904, Springer-Verlag, 1995.
- [11] Gold, E. M.: Language Identification in the Limit, *Information and Control*, **10**, 1967, 447–474.
- [12] Hopcroft, J., Ullman, J.: *Introduction to Automata Theory, Languages, and Computation*, Addison-Wesley, 1979.
- [13] Jain, S., Kinber, E.: Intrinsic Complexity of Learning Geometrical Concepts from Positive Data, *Journal of Computer and System Sciences*, **67**, 2003, 546–607.
- [14] Jain, S., Kinber, E., Wiehagen, R.: Language Learning from Texts: Degrees of Intrinsic Complexity and Their Characterizations, *Journal of Computer and System Sciences*, **63**, 2001, 305–354.

- [15] Jain, S., Osherson, D., Royer, J., Sharma, A.: *Systems that Learn: An Introduction to Learning Theory*, second edition, MIT Press, Cambridge, Mass., 1999.
- [16] Jain, S., Sharma, A.: Characterizing Language Learning by Standardizing Operations, *Journal of Computer and System Sciences*, **49**(1), 1994, 96–107.
- [17] Jain, S., Sharma, A.: The Intrinsic Complexity of Language Identification, *Journal of Computer and System Sciences*, **52**, 1996, 393–402.
- [18] Jain, S., Sharma, A.: Elementary formal systems, intrinsic complexity, and procrastination, *Information and Computation*, **132**, 1997, 65–84.
- [19] Jain, S., Sharma, A.: The Structure of Intrinsic Complexity of Learning, *Journal of Symbolic Logic*, **62**, 1997, 1187–1201.
- [20] Kinber, E.: On Comparison of Limit Identification and Limit Standardization of General Recursive Functions, *Uch. zap. Latv. univ.*, **233**, 1975, 45–56.
- [21] Kinber, E.: Monotonicity versus Efficiency for Learning Languages from Texts, *Algorithmic Learning Theory: Fourth International Workshop on Analogical and Inductive Inference (AII '94) and Fifth International Workshop on Algorithmic Learning Theory (ALT '94)* (S. Arikawa, K. Jantke, Eds.), 872, Springer-Verlag, 1994.
- [22] Kinber, E., Papazian, C., Smith, C., Wiehagen, R.: On the Intrinsic Complexity of Learning Infinite Objects from Finite Samples, *Proceedings of the Twelfth Annual Conference on Computational Learning Theory*, ACM Press, 1999.
- [23] Kinber, E., Stephan, F.: Language Learning from Texts: Mind Changes, Limited Memory and Monotonicity, *Information and Computation*, **123**, 1995, 224–241.
- [24] Lange, S., Zeugmann, T.: Learning Recursive Languages With a Bounded Number of Mind Changes, *International Journal of Foundations of Computer Science*, **4**(2), 1993, 157–178.
- [25] Machtey, M., Young, P.: *An Introduction to the General Theory of Algorithms*, North Holland, New York, 1978.
- [26] Motoki, T., Shinohara, T., Wright, K.: The Correct Definition of Finite Elasticity: Corrigendum to Identification of Unions, *Proceedings of the Fourth Annual Workshop on Computational Learning Theory* (L. Valiant, M. Warmuth, Eds.), Morgan Kaufmann, 1991.
- [27] Osherson, D., Weinstein, S.: Criteria of Language Learning, *Information and Control*, **52**, 1982, 123–138.
- [28] Rogers, H.: *Theory of Recursive Functions and Effective Computability*, McGraw-Hill, 1967, Reprinted, MIT Press 1987.
- [29] Wiehagen, R.: Identification of Formal languages, *Mathematical Foundations of Computer Science*, **53**, Springer-Verlag, 1977.
- [30] Wiehagen, R.: On the Complexity of Program Synthesis from Examples, *Journal of Information Processing and Cybernetics (EIK)*, **22**, 1986, 305–323.
- [31] Wright, K.: Identification of Unions of Languages Drawn from an Identifiable Class, *Proceedings of the Second Annual Workshop on Computational Learning Theory* (R. Rivest, D. Haussler, M. Warmuth, Eds.), Morgan Kaufmann, 1989.
- [32] Zeugmann, T., Lange, S.: A Guided Tour Across the Boundaries of Learning Recursive Languages, in: *Algorithmic Learning for Knowledge-Based Systems* (K. Jantke, S. Lange, Eds.), vol. 961 of *Lecture Notes in Artificial Intelligence*, Springer-Verlag, 1995, 190–258.