九州大学学術情報リポジトリ
Kyushu University Institutional Repository

Inductive Inference Machines That Can Refute Hypothesis Spaces

Mukochi, Yasuhito
Research Institute of Fundamental Information Science Kyushu University

Arikawa, Setsuo
Research Institute of Fundamental Information Science Kyushu University

https://hdl.handle.net/2324/3169

出版情報：RIFIS Technical Report. 67, 1993-03-27. Research Institute of Fundamental Information Science, Kyushu University
バージョン：
権利関係：

RIFIS-TR-CS-67

RIFIS Technical Report

Inductive Inference Machines That Can Refute Hypothesis Spaces

Yasuhito Mukouchi
Setsuo Arikawa

March 27, 1993
Revised: January 15, 1994

Research Institute of Fundamental Information Science
Kyushu University 33
Fukuoka 812, Japan

E-mail: mukouchi@rifis.sci.kyushu-u.ac.jp Phone: 092-641-1101 ex. 4470

INDUCTIVE INFERENCE MACHINES THAT CAN REFUTE HYPOTHESIS SPACES*

Yasuhito MUKOUCHI and Setsuo ARIKAWA

Research Institute of Fundamental Information Science
Kyushu University 33, Fukuoka 812, Japan

Abstract. This paper intends to give a theoretical foundation of machine discovery from examples. We point out that the essence of a computational logic of scientific discovery or a logic of machine discovery is the refutability of the entire spaces of hypotheses. We discuss this issue in the framework of inductive inference of length-bounded elementary formal systems (EFS's, for short), which are a kind of logic programs over strings of characters and correspond to context-sensitive grammars in Chomsky hierarchy.

We first present some characterization theorems on inductive inference machines that can refute hypothesis spaces. Then we show differences between our inductive inference and some other related inferences such as in the criteria of reliable identification, finite identification and identification in the limit. Finally we show that for any n, the class, i.e. hypothesis space, of length-bounded EFS's with at most n axioms is inferable in our sense, that is, the class is refutable by a consistently working inductive inference machine. This means that sufficiently large hypothesis spaces are identifiable and refutable.

> A scientist, whether theorist or experimenter, puts forward statements, or systems of statements, and test them step by step. In the field of the empirical sciences, more particularly, he constructs hypotheses, or systems of theories, and test them against experience by observation and experiment.
>
> *Karl R. Popper: The Logic of Scientific Discovery*

1. Introduction

In the middle of this century the logic of scientific discovery was deeply discussed by philosophers[23, 24]. Recently in Artificial Intelligence, especially in Cognitive Science, researchers are extensively discussing frameworks for scientific discovery from various viewpoints[33]. They have obtained a lot of rich results on the components of scientific behavior such as scientific knowledge structures and scientific activities. However they look little dependent on the philosophical results.

*This work is partly supported by Grants-in-Aid for Scientific Research on Priority Areas, "Knowledge Science" and "Genome Informatics" from the Ministry of Education, Science and Culture, Japan.

Before going into such detailed discussions we need to set up a computational logic of scientific discovery in a mathematical way so that we can precisely discuss what kinds of machine discovery can work. One of the best ways to this should be to reexamine the philosophical results from computational viewpoints. In the present paper we start with making the Popperian logic of scientific discovery computational.

The logic of scientific discovery is essentially a triad of problem discovery, solution invention and critical test[35]. Popper was mainly interested in the final stage but not so much in the first two stages. More exactly the Popperian logic of scientific discovery concentrated on the testability, falsifiability or refutability of hypotheses or scientific theories. He also asserted that scientific theory should have been refuted by observed facts and any such theory could by no means be verified[23]. Thus we tentatively believe the current theory until we face with an observation which is inconsistent with the theory.

His assertion had an influence on the studies of inductive inference started with Gold's identification in the limit, some of which is called Popperian induction[3]. The consistent and conservative inductive inference can be viewed as a computational realization of the Popperian notion of refutability. In the inductive inference, the inference machine requires data or facts from time to time and produces hypotheses from time to time. The hypotheses produced by the machine are to be consistent with the facts read so far, and each of them is to be refuted when the machine faces with inconsistent data or facts.

Thus the Popperian logic of scientific discovery can be viewed as a basis of the modern inductive inference studies. The inductive inference is thus a mathematical basis of machine learning. *Then what should be a logic of machine discovery or a computational logic of scientific discovery?*

The machine discovery we are concerned with in this paper is to make computers discover some scientific theories from given data or facts. Hence machine learning should be a key technology for machine discovery. In fact, by using machine learning techniques many results have been reported by many authors[33]. In machine learning first we must select a hypothesis space from which the learning machine proposes theories or hypotheses. The space is naturally required to be large, but to make the learning efficient it is required to be small. As far as data or facts are presented according to a hypothesis that is unknown but guaranteed to be in the space as in the ordinal inductive inference, the machine will eventually identify the hypothesis, and hence no problem may arise. In machine discovery, however, we can not assume this. God knows whether or not a hypothesis behinds the data or facts belongs to the space.

If the hypothesis is not in the space, most learning machines will continue for ever to search the space for a new hypothesis. Usually we can not know the time when to stop such an ineffective searching. This is the most crucial problem we must solve in realizing machine discovery systems. In machine discovery the sequences of data or facts are given at first independently of the space. We can not give in advance the space that includes the desired theory. As stated above, however, we have to keep the space as small as possible without any guarantee that the theory is in it. If the learning machine can explicitly tell us that there are no theories in the space which explain the given sequence, the machine will work for machine discovery.

Hence the essence of a computational logic of scientific discovery should be that the entire hypothesis space is refutable by a sequence of observed data or facts. If there exist rich hypothesis spaces that can be refuted, we can give a space and a sequence to the

machine, and then we can just wait for an output from it. The machine will discover a hypothesis which is producing the sequence if it is in the space, otherwise it will refute the whole of the space and stop. When the space is refuted, we may give another space to the machine and try to make such a discovery in the new space.

In this paper we choose the inductive inference as the framework for machine learning. Then the machine discovery system is an inductive inference machine that can refute hypothesis spaces. In order to make our discussion clearer we also choose, as the hypothesis spaces, classes of elementary formal systems (EFS's, for short) which are a kind of logic programs over strings of characters. Thus we are concerned with formal languages in this paper. Moreover we assume as in the ordinal inductive inference of languages that every class in question be an indexed family of recursive languages. This assumption is quite natural to make a grammar, i.e. a hypothesis or theory, refutable by an observation and also to generate grammars as hypotheses automatically and successively. Note here that any indexed family of recursive languages is identifiable in the limit from complete data, i.e. positive and negative data, but the class of all recursive languages is not such a family. Hence it is reasonable to choose some subclasses of the recursive languages as the hypothesis spaces for machine learning and machine discovery.

If the class is a finite set of recursive languages, it is trivially refutable from complete data. Also if the class contains all finite languages, it is easily shown not to be refutable. *Then are there any meaningful classes, i.e. hypothesis spaces, that are identifiable and refutable?* We give a positive answer to this question. We will say such classes to be *refutably inferable.*

We first show some characterizations of such inductive inference. Then we show that some sufficiently large classes of formal languages are refutably inferable from complete data, while the classes that are refutably inferable only from positive data are very small.

This paper is organized as follows: Section 2 presents basic notions and definitions necessary for our discussion. In Section 3 we discuss some conditions on refutable inferability from positive data or complete data. Concerning refutable inferability from *positive data*, we present some necessary and sufficient conditions and reveal that the power is very small. Then in Section 4 we show the differences between the inferable classes in the criteria of refutable identification, reliable identification, finite identification and identification in the limit. Among them the reliable identification is only the inference that deals with sequences from the hypotheses not in the given spaces. However, as we are seeing in the next section, the reliable inference machine does not tell us that the sequence is not in the space, but it just does not converge to any of hypothesis in the space. In Section 5 and Section 6 we presents some natural classes that are refutably inferable from complete data. First we show that a class which consists of unions of at most n concepts from n classes is refutably inferable from complete data, if each class satisfies a certain condition. Then we show our main result: The classes definable by length-bounded EFS's with at most n axioms are refutably inferable from complete data, and reveal that there are sufficiently large classes that are refutably inferable from complete data.

2. Preliminaries

We start with basic definitions and notions on inductive inference of indexed families of recursive concepts.

Let U be a recursively enumerable set to which we refer as a *universal set.* Then we call $L \subseteq U$ a *concept.* In case the universal set U is the set Σ^+ of all nonnull finite strings over a finite alphabet Σ, we also call $L \subseteq U$ a *language.*

Definition. Let $N = \{1, 2, \cdots\}$ be the set of all natural numbers. A class $\mathcal{C} = \{L_i\}_{i \in N}$ of concepts is said to be an *indexed family of recursive concepts*, if there is a recursive function $f : N \times U \to \{0, 1\}$ such that

$$f(i, w) = \begin{cases} 1, & \text{if } w \in L_i, \\ 0, & \text{otherwise.} \end{cases}$$

In what follows, we assume that a class of concepts is an indexed family of recursive concepts without any notice, and identify a class with a hypothesis space.

Definition. A *positive presentation*, or a *text*, of a *nonempty* concept L is an infinite sequence $w_1, w_2, \cdots$ of elements in the universal set U such that $\{w_1, w_2, \cdots\} = L$. A *complete presentation*, or an *informant*, of a concept L is an infinite sequence $(w_1, t_1), (w_2, t_2), \cdots$ of elements in $U \times \{+, -\}$ such that $\{w_i \mid t_i = +, i \geq 1\} = L$ and $\{w_i \mid t_i = -, i \geq 1\} = L^c (= U \setminus L)$. In what follows, σ or δ denotes a positive or complete presentation, and $\sigma[n]$ denotes the σ's initial segment of length $n \geq 0$. For a positive or complete presentation σ, each element in σ is called a *fact.* For a positive presentation σ, $\sigma[n]^+$ denotes the set of all facts in $\sigma[n]$. For a complete presentation σ, $\sigma[n]^+$ (resp., $\sigma[n]^-$) denotes the set of all elements in the universal set U that appear in $\sigma[n]$ with the sign '+' (resp., the sign '−'), that is, $\sigma[n]^+ = \{w_i \mid (w_i, +) \in \sigma[n]\}$ and $\sigma[n]^- = \{w_i \mid (w_i, -) \in \sigma[n]\}$.

A set T is said to be *consistent* with a concept L, if $T \subseteq L$. A pair (T, F) of sets is said to be *consistent* with a concept L, if $T \subseteq L$ and $F \subseteq L^c$. For a positive presentation σ and for $n \geq 0$, the finite sequence $\sigma[n]$ is said to be *consistent* with a concept L, if $\sigma[n]^+ \subseteq L$. For a complete presentation σ and for $n \geq 0$, the finite sequence $\sigma[n]$ is said to be *consistent* with a concept L, if $\sigma[n]^+ \subseteq L$ and $\sigma[n]^- \subseteq L^c$.

For two sequences ψ_1 and ψ_2, the sequence which is obtained by concatenating ψ_1 with ψ_2 is denoted by $\psi_1 \cdot \psi_2$.

Here we note that for a class $\mathcal{C} = \{L_i\}_{i \in N}$ and explicitly given finite sets $T, F \subseteq U$, whether $T \subseteq L_i$ or not and whether (T, F) is consistent with L_i or not are recursively decidable for any index i, because for any $w \in U$, whether $w \in L_i$ or not is recursively decidable.

Definition. An *inductive inference machine* (*IIM*, for short) is an effective procedure, or a certain type of Turing machine, which requests inputs from time to time and produces positive integers from time to time. An *inductive inference machine that can refute hypothesis spaces* (*RIIM*, for short) is an effective procedure, or a certain type of Turing machine, which requests inputs from time to time and either (i) produces positive integers from time to time or (ii) refutes the class and stops after producing some positive integers. The outputs produced by a machine are called *guesses*.

For an IIM or an RIIM M and a nonempty initial segment $\sigma[n] = w_1, w_2, \cdots, w_n$ of a positive or complete presentation, we define $M(\sigma[n])$ as follows: Initialize and start M. If it requests a fact for the i-th time with $1 \leq i \leq n$, then feed w_i and continue the execution.

(I) In case M requests the $(n+1)$-st fact, or it stops after it requested the n-th fact. If it produces a positive integer or the 'refutation' sign after it requested the n-th fact, then let $M(\sigma[n])$ be the last integer or the 'refutation' sign produced by M, otherwise let $M(\sigma[n]) = 0$.

(II) In case M stops before requesting the n-th fact. Let $M(\sigma[n]) = 0$.

(III) Otherwise. Leave $M(\sigma[n])$ undefined.

For an IIM or an RIIM M, we define $\overline{M}(\sigma[n])$ as follows:

(I) In case $M(\sigma[n])$ is defined. (i) If there is the 'refutation' sign in the sequence $M(\sigma[1]), M(\sigma[2]), \cdots, M(\sigma[n])$, then let $\overline{M}(\sigma[n])$ be the 'refutation' sign. (ii) Otherwise. If there is a positive integer in the sequence $M(\sigma[1]), M(\sigma[2]), \cdots, M(\sigma[n])$, then let $\overline{M}(\sigma[n])$ be the last positive integer in the sequence, otherwise let $\overline{M}(\sigma[n]) = 0$.

(II) Otherwise. Leave $\overline{M}(\sigma[n])$ undefined.

For an IIM or an RIIM M, it is easy to see that for any $n \geq 1$, if $M(\sigma[n])$ is defined, then $M(\sigma[1]), M(\sigma[2]), \cdots, M(\sigma[n-1])$ are also defined.

The intended interpretation is as follows: (i) In case $M(\sigma[n])$ is defined as the 'refutation' sign, M refutes the class concerned. (ii) In case $M(\sigma[n])$ is defined as a positive integer i, M guesses the i-th concept in the class. (iii) In case $M(\sigma[n])$ is defined as the integer 0, M makes no guess. (iv) In case $M(\sigma[n])$ is undefined, M is out of control.

For two nonempty finite sequences ψ_1 and ψ_2, we write $M(\psi_1) = M(\psi_2)$, if (i) both $M(\psi_1)$ and $M(\psi_2)$ are undefined, or (ii) both $M(\psi_1)$ and $M(\psi_2)$ are defined and their values are identical. For a nonempty finite sequence ψ and an integer i, we write $M(\psi) = i$ (resp., $M(\psi) > i$), if $M(\psi)$ is defined as an integer and the value of $M(\psi)$ is equal to i (resp., greater than i). In a similar way, we also define the relations $\overline{M}(\psi_1) = \overline{M}(\psi_2)$, $\overline{M}(\psi) = i$ and $\overline{M}(\psi) > i$.

Hereafter, for a concept $L \subseteq U$, we write $L \in \mathcal{C}$, if there is an $L_i \in \mathcal{C}$ such that $L_i = L$.

Definition. An IIM or an RIIM M is said to *converge to an index i for a positive or complete presentation σ*, if there is an $n \geq 1$ such that for any $m \geq n$, $\overline{M}(\sigma[m]) = i$.

An RIIM M is said to *refute a class $\mathcal{C}$ from a positive or complete presentation σ*, if there is an $n \geq 1$ such that $\overline{M}(\sigma[n])$ is the 'refutation' sign. In this case we also say that M refutes the class $\mathcal{C}$ from $\sigma[n]$.

Let $\mathcal{C} = \{L_i\}_{i \in N}$ be a class. For a concept $L_i \in \mathcal{C}$ and a positive or complete presentation σ of L_i, an IIM M or an RIIM M is said to *infer the concept L_i w.r.t. $\mathcal{C}$ in the limit from σ*, if M converges to an index j with $L_j = L_i$ for σ.

(a) An IIM M is said to *infer a class $\mathcal{C} = \{L_i\}_{i \in N}$ in the limit from positive data* (resp., *complete data*), if for any $L_i \in \mathcal{C}$, M infers L_i w.r.t. $\mathcal{C}$ in the limit from any positive presentation σ (resp., any complete presentation σ) of L_i. A class $\mathcal{C}$ is said to be *inferable in the limit from positive data* (resp., *complete data*), if there is an IIM M which infers the class $\mathcal{C}$ in the limit from positive data (resp., complete data).

(b) An IIM M is said to *reliably infer a class $\mathcal{C}$ from positive data* (resp., *complete data*), if it satisfies the following condition: For any nonempty concept L (resp., any concept L) and any positive presentation σ (resp., any complete presentation σ) of L, (i) if $L \in \mathcal{C}$, then M infers L w.r.t. $\mathcal{C}$ in the limit from σ, (ii) otherwise M does not converge to any index for

σ. A class $\mathcal{C}$ is said to be *reliably inferable from positive data* (resp., *complete data*), if there is an IIM M which reliably infers the class $\mathcal{C}$ from positive data (resp., complete data).

(c) An RIIM M is said to *refutably infer a class $\mathcal{C}$ from positive data* (resp., *complete data*), if it satisfies the following condition: For any nonempty concept L (resp., any concept L) and any positive presentation σ (resp., any complete presentation σ) of L, (i) if $L \in \mathcal{C}$, then M infers L w.r.t. $\mathcal{C}$ in the limit from σ, (ii) otherwise M refutes the class $\mathcal{C}$ from σ. A class $\mathcal{C}$ is said to be *refutably inferable from positive data* (resp., *complete data*), if there is an RIIM M which refutably infers the class $\mathcal{C}$ from positive data (resp., complete data).

Note that when we consider inductive inference from positive data, we restrict every concept to a *nonempty* concept, because we can not make any positive presentation of the empty concept.

The notion of reliable inference was introduced by Minicozzi[20] and Blum&Blum[11] for function learning, and it was adapted to language learning by Sakurai[26]. By definition, it is easy to see that if a class $\mathcal{C}$ is *refutably* inferable from positive data (resp., complete data), then $\mathcal{C}$ is *reliably* inferable from positive data (resp., complete data). However the converse does not hold as shown in Section 4.

Note that if an inference machine M does not converge to any index for a positive or complete presentation σ, then $\overline{M}(\sigma[n])$ may be undefined for some $n \geq 1$. On the other hand, we implicitly use the following Proposition 1 in showing some properties on inferability.

Proposition 1. *(a) Assume that an IIM M infers a class $\mathcal{C}$ in the limit from* positive data. *Then for any nonempty finite sequence ψ consisting of elements in U, if there exists an $L_i \in \mathcal{C}$ such that all elements in ψ are in L_i, then $\overline{M}(\psi)$ is always defined.*

(b) Assume that an RIIM M refutably infers a class $\mathcal{C}$ from positive data. *Then for any nonempty finite sequence ψ consisting of elements in U, $\overline{M}(\psi)$ is always defined.*

Similar statements are also valid for the case of complete data.

The above Proposition 1 claims that as far as we feed facts that are from a certain concept in the class, an IIM either (i) successively requests another facts in a finite time forever or (ii) stops in a finite time after producing some guesses. On the other hand, even when we feed *any* facts that may not be from any concept in the class, an RIIM either (i) successively requests another facts in a finite time forever or (ii) stops in a finite time after producing some guesses.

Since we are considering an indexed family of recursive concepts, every class can be inferred from complete data by a simple enumerative method. However we can not take the class of all recursive concepts as a hypothesis space, because the following Proposition 2 holds.

Proposition 2. *The class $\mathcal{C}$ of* all *recursive concepts is not an indexed family of recursive concepts.*

In case an RIIM M is fed a positive or complete presentation of a non-recursive concept, M should refute the class. Therefore even if we could take the class of all recursive concepts, it would be still significant to consider refutable inferability.

For reliable inferability, the following Theorem 3 holds.

Theorem 3. *(a) A class $\mathcal{C}$ is reliably inferable from positive data, if and only if $\mathcal{C}$ contains no infinite concept (Sakurai[26]).*

(b) Every class is reliably inferable from complete data.

In this paper, an effective procedure is said to *recursively generate* a finite set T, if it enumerates all elements in T and *then halts.* An effective procedure P is said to be *uniformly and recursively generate* a finite-set-valued function F with parameters $x_1, \cdots, x_n$, if P on any input $(x_1, \cdots, x_n)$ in the domain of F recursively generates the finite set $F(x_1, \cdots, x_n)$. A finite-set-valued function F with parameters $x_1, \cdots, x_n$ is said to be *uniformly and recursively generable*, if there is an effective procedure which uniformly and recursively generates the function F.

3. Characterizations

In order to characterize the refutable inferability, we need the following Lemma 4.

Lemma 4. *Let M be an RIIM which refutably infers a class $\mathcal{C}$ from positive data (resp., complete data). Then for a nonempty concept L (resp., a concept L), for a positive presentation σ (resp., a complete presentation σ) of L and for $n \geq 1$, if M refutes the class $\mathcal{C}$ from $\sigma[n]$, then $\sigma[n]$ is* not *consistent with any $L_i \in \mathcal{C}$.*

Proof. Assume that an RIIM M refutes a class $\mathcal{C}$ from $\sigma[n]$. Then suppose that there is an $L_i \in \mathcal{C}$ such that $\sigma[n]$ is consistent with L_i. Let δ be a positive presentation (resp., a complete presentation) of L_i. Then the infinite sequence $\sigma[n] \cdot \delta$ becomes a positive presentation (resp., a complete presentation) of L_i. Therefore M can not infer L_i w.r.t. $\mathcal{C}$ in the limit from $\sigma[n] \cdot \delta$, which contradicts the assumption. □

By the above Lemma 4, we obtain the following Proposition 5.

Proposition 5. *(a) If a class $\mathcal{C}$ is refutably inferable from positive data, then*

(3.1) for any nonempty concept $L \notin \mathcal{C}$, there is a finite set $T \subseteq L$ such that T is not *consistent with any $L_i \in \mathcal{C}$.*

(b) If a class $\mathcal{C}$ is refutably inferable from complete data, then

(3.2) for any concept $L \notin \mathcal{C}$, there are finite sets $T \subseteq L$ and $F \subseteq L^c$ such that (T, F) is not *consistent with any $L_i \in \mathcal{C}$.*

Proof. We only give the proof of (a). The proof of (b) can be given in a similar way.

Assume that an RIIM M refutably infers a class $\mathcal{C}$ from positive data. Let $L \notin \mathcal{C}$ be a nonempty concept, and let σ be an arbitrary positive presentation of L. By definition, there is an $n \geq 1$ such that M refutes the class $\mathcal{C}$ from $\sigma[n]$. Let $T = \sigma[n]^+$. Then by Lemma 4, T is *not* consistent with any $L_i \in \mathcal{C}$. □

Corollary 6. *If a class $\mathcal{C}$ contains all nonempty finite concepts, then $\mathcal{C}$ is* not *refutably inferable from positive data or complete data.*

Proof. We only give the proof of the case of complete data. The proof for positive data can be given in a similar way.

Assume that a class $\mathcal{C}$ contains all nonempty finite concepts. Then let $L \notin \mathcal{C}$ be a concept. Let $T \subseteq L$ and $F \subseteq L^c$ be finite sets. (i) In case T is not empty. There is an $L_i \in \mathcal{C}$ with $T = L_i$, and it follows that (T, F) is consistent with L_i. (ii) In case T is empty. Since F is a finite set, there is an $L_i \in \mathcal{C}$ such that $F \subseteq L_i^c$, which means (T, F) is consistent with L_i. Therefore by Proposition 5, we see that $\mathcal{C}$ is not refutably inferable from complete data. □

In characterizing the refutable inferability, the notion of consistency plays an important role.

Definition. Let $\mathcal{C} = \{L_i\}_{i \in N}$ be a class. An RIIM M which refutably infers a class $\mathcal{C}$ from positive data (resp., complete data) is said to be *consistently working*, if it satisfies the following condition: For any nonempty concept L (resp., any concept L), any positive presentation σ (resp., any complete presentation σ) of L and any $n \geq 1$, (i) if $M(\sigma[n])$ is the 'refutation' sign, then $\sigma[n]$ is *not* consistent with any $L_i \in \mathcal{C}$, (ii) if $M(\sigma[n]) > 0$, then $\sigma[n]$ is consistent with $L_{M(\sigma[n])}$.

An RIIM M which refutably infers a class $\mathcal{C}$ from positive data (resp., complete data) is said to be *responsively working*, if it satisfies the following condition: For any nonempty concept L (resp., any concept L), any positive presentation σ (resp., any complete presentation σ) of L and any $n \geq 1$, if M does not refute the class $\mathcal{C}$ from $\sigma[n]$, then $M(\sigma[n]) > 0$ holds, that is, while M does not refute the class, M produces a guess between any two input requests in the computation of M on input σ.

A class $\mathcal{C}$ is said to be *refutably, consistently and responsively inferable* from positive data (resp., complete data), if there is a consistently and responsively working RIIM which refutably infers the class $\mathcal{C}$ from positive data (resp., complete data).

In a similar way, we can also define a consistently or responsively working IIM (cf. Angluin[2]).

Here we note that a consistently and responsively working RIIM refutes a class immediately after the observed data become *not* consistent with any concept in the class.

Since we are considering an indexed family of recursive concepts, we can easily show that if a class $\mathcal{C}$ is inferable in the limit from positive data or complete data, then it can be achieved by a *consistently and responsively* working IIM. Furthermore, as shown later, if a class $\mathcal{C}$ is refutably inferable from positive data or complete data, then it can be achieved by a *consistently and responsively* working RIIM.

Definition. For a finite set $T \subseteq U$, let

$$\mathrm{econs}_p(T) = \begin{cases} 1, & \text{if there exists an } L_i \in \mathcal{C} \text{ such that} \\ & \qquad T \text{ is consistent with } L_i, \\ 0, & \text{otherwise.} \end{cases}$$

For finite sets $T, F \subseteq U$, let

$$\mathrm{econs}_c(T, F) = \begin{cases} 1, & \text{if there exists an } L_i \in \mathcal{C} \text{ such that} \\ & \qquad (T, F) \text{ is consistent with } L_i, \\ 0, & \text{otherwise.} \end{cases}$$

Proposition 7. *If a class $\mathcal{C}$ is refutably inferable from positive data, then*

(3.3) the function econs_p *for $\mathcal{C}$ is recursive.*

Proof. Assume that an RIIM M refutably infers $\mathcal{C}$ from positive data. Let $T = \{w_1, \cdots, w_n\} \subseteq U$ be a nonempty finite set, and put $\sigma = w_1, w_2, \cdots, w_n, w_1, w_1, w_1, \cdots$. Clearly, the infinite sequence σ is a positive presentation of the concept T. Thus when we successively feed σ, M either refutes the class $\mathcal{C}$ or produces an index i with $T = L_i$ after producing some positive integers. Therefore we can recursively compute the function $\mathrm{econs}_p(T)$ for $\mathcal{C}$ as follows: Simulate M with presenting σ. During the simulation, (i) if M refutes the class $\mathcal{C}$, then output 0 and stop, (ii) if M produces an index i with $T \subseteq L_i$, then output 1 and stop, (iii) otherwise continue the simulation. We note that whether $T \subseteq L_i$ or not is recursively decidable. By Lemma 4, it is clear that the above output agrees with the $\mathrm{econs}_p(T)$. □

Theorem 8 (Based on Kinber[14]). *If a class $\mathcal{C}$ is refutably inferable from complete data, then*

(3.4) the function econs_c *for $\mathcal{C}$ is recursive.*

Proof. Assume that an RIIM M refutably infers $\mathcal{C}$ from complete data. Let $T = \{w_1, \cdots, w_n\} \subseteq U$ and $F = \{w_{n+1}, \cdots, w_m\} \subseteq U$ be finite sets.

It is easy to see that if $T \cap F \neq \phi$, then $\mathrm{econs}_c(T, F) = 0$. Thus in what follows, we assume $T \cap F = \phi$.

Let $\psi_0 = (w_1, +), \cdots, (w_n, +), (w_{n+1}, -), \cdots, (w_m, -)$, and let $u_1, u_2, \cdots$ be an effective enumeration of $U \setminus (T \cup F)$. Then let $\mathcal{T}$ be the set of all initial segments of $u_1, u_2, \cdots$ coupled with $+$ and $-$, that is, $\mathcal{T} = \{\phi;\ (u_1, +);\ (u_1, -);\ (u_1, +), (u_2, +);\ (u_1, +), (u_2, -);\ (u_1, -), (u_2, +);\ (u_1, -), (u_2, -);\ \cdots\}$. We define the binary relation $\sqsubseteq$ over $\mathcal{T}$ as follows: $\psi_1 \sqsubseteq \psi_2$ if and only if ψ_1 is an initial segment of ψ_2. This gives a partial ordering of $\mathcal{T}$, and it becomes a binary tree, which can be diagrammed in Figure 1.

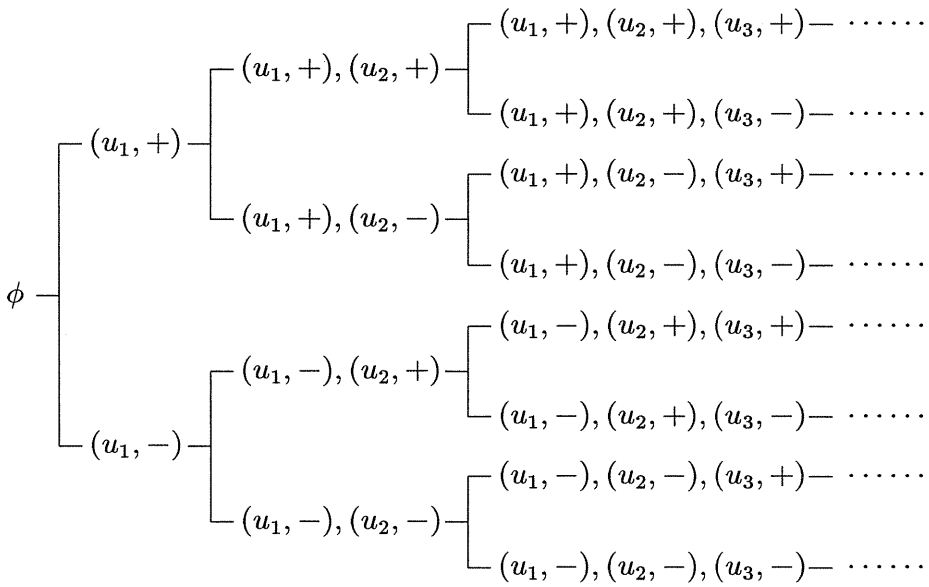

Figure 1: The binary tree $\mathcal{T}$

Then we define a subtree $\mathcal{S}$ of $\mathcal{T}$ as follows:

$$\mathcal{S} = \{\psi \in \mathcal{T} \mid \overline{M}(\psi_0 \cdot \psi) \neq \text{'refutation'}\}.$$

Here we note that if $\psi_1 \sqsubseteq \psi_2$ and $\overline{M}(\psi_0 \cdot \psi_1) =$ 'refutation', then $\overline{M}(\psi_0 \cdot \psi_2) =$ 'refutation'.

Claim A: The subtree $\mathcal{S}$ is finite, if and only if there is no concept $L_i \in \mathcal{C}$ such that (T, F) is consistent with L_i.

Proof of the claim. (I) The 'if' part. Assume that there is no concept $L_i \in \mathcal{C}$ such that (T, F) is consistent with L_i. Then suppose that the subtree $\mathcal{S}$ has an infinite branch, say $\psi_1, \psi_2, \cdots$. By the construction of the subtree $\mathcal{S}$, there is an infinite sequence $t_1, t_2, \cdots \in \{+, -\}$ such that for any $i \geq 1$, $\psi_i = (u_1, t_1), (u_2, t_2), \cdots, (u_i, t_i)$. Put $\sigma = \psi_0, (u_1, t_1), (u_2, t_2), \cdots$, and let $L = \{u_i \in U \mid (u_i, +) \in \sigma,\ i \geq 1\}$ be a concept. Then (T, F) is consistent with L, and σ is a complete presentation of L. By assumption, L is not in $\mathcal{C}$, and it follows that M refutes $\mathcal{C}$ from $\sigma[n]$ for some $n \geq 1$. However, by the construction, there is a $j \geq 1$ such that $\sigma[n] = \psi_0 \cdot \psi_j$. This contradicts the assumption of $\psi_j \in \mathcal{S}$.

Thus we see that the subtree $\mathcal{S}$ has no infinite branch, and it follows by Endlichkeits-lemma for trees with finite branching (cf. e.g. Rogers[25], Exercise 9.40) that the subtree $\mathcal{S}$ is finite.

(II) The 'only if' part. Assume that the subtree $\mathcal{S}$ is finite. Therefore the subtree $\mathcal{S}$ has no infinite branch. Then suppose that there is an $L_i \in \mathcal{C}$ such that (T, F) is consistent with L_i. For $j \geq 1$, let $t_j =$ '+' if $u_j \in L_i$, otherwise let $t_j =$ '$-$'. Then put $\sigma = \psi_0, (u_1, t_1), (u_2, t_2), \cdots$. By the construction, σ is a complete presentation of L_i. Thus M does not refute $\mathcal{C}$ from σ. It is easy to see that this contradicts the assumption. ■

Claim B: There is an $L_i \in \mathcal{C}$ such that (T, F) is consistent with L_i, if and only if there is a $\psi \in \mathcal{S}$ such that $\overline{M}(\psi_0 \cdot \psi) > 0$ and (T, F) is consistent with $L_{\overline{M}(\psi_0 \cdot \psi)}$.

Proof of the claim. The 'if' part is obvious. Thus we only give the proof of the 'only if' part. Assume that there is an $L_i \in \mathcal{C}$ such that (T, F) is consistent with L_i. For $j \geq 1$, let $t_j =$ '+' if $u_j \in L_i$, otherwise let $t_j =$ '$-$'. Then put $\sigma = \psi_0, (u_1, t_1), (u_2, t_2), \cdots$. By the construction, σ is a complete presentation of L_i. Since M infers L_i w.r.t. $\mathcal{C}$ in the limit from σ, it follows that there is an $n \geq 1$ such that $\overline{M}(\sigma[n]) > 0$ and $L_i = L_{\overline{M}(\sigma[n])}$. It is easy to see that there is a $\psi \in \mathcal{S}$ such that $\psi_0 \cdot \psi = \sigma[n]$. ■

Therefore we can compute $\mathrm{econs}_c(T, F)$ as follows: Search for a node $\psi \in \mathcal{S}$ such that $\overline{M}(\psi_0 \cdot \psi) > 0$ and (T, F) is consistent with $L_{\overline{M}(\psi_0 \cdot \psi)}$. By Claim A and B, we see that there must happen one of two cases:

(1) Such a node is found.

(2) The subtree $\mathcal{S}$ is confirmed to be finite.

In case (1), put $\mathrm{econs}_c(T, F) = 1$ and in case (2), put $\mathrm{econs}_c(T, F) = 0$. It is easy to see that the obtained results agrees with the definition of $\mathrm{econs}_c(T, F)$. □

Theorem 9. *(a) If a class $\mathcal{C}$ satisfies the following three conditions, then $\mathcal{C}$ is refutably, consistently and responsively inferable from positive data.*

(3.1) For any nonempty concept $L \notin \mathcal{C}$, there is a finite set $T \subseteq L$ such that T is not *consistent with any $L_i \in \mathcal{C}$.*

(3.3) The function econs_p *for $\mathcal{C}$ is recursive.*

(3.5) The class $\mathcal{C}$ is inferable in the limit from positive data.

(b) If a class $\mathcal{C}$ satisfies the following two conditions, then $\mathcal{C}$ is refutably, consistently and responsively inferable from complete data.

(3.2) For any concept $L \notin \mathcal{C}$, there are finite sets $T \subseteq L$ and $F \subseteq L^c$ such that (T, F) is not *consistent with any $L_i \in \mathcal{C}$.*

(3.4) The function econs_c *for $\mathcal{C}$ is recursive.*

Proof. We only give the proof of (a). The proof of (b) can be given in a similar way, where we note that every indexed family of recursive concepts is consistently and responsively inferable in the limit from complete data.

Assume that a class $\mathcal{C}$ satisfies the above three conditions (3.1), (3.3) and (3.5). Let M be an IIM which infers $\mathcal{C}$ in the limit from positive data. Without loss of generality, we can assume that it works consistently and responsively. Then let us consider the procedure in Figure 2.

```
Procedure RIIM M;
begin
    T = φ;
    repeat
        read the next fact w and store it in T;
        if econs_p(T) = 0 then
            refute the class C and stop;
        else begin
            simulate M with presenting the fact w until requesting the next fact;
            if M produces a guess then output it;
        end;
    forever;
end.
```

Figure 2: An inference machine that can refute hypothesis spaces from positive data

Assume that we feed a positive presentation σ of a nonempty concept L to the procedure.

(I) In case $L \notin \mathcal{C}$. By the condition (3.1), there is a finite set $T \subseteq L$ such that $\mathrm{econs}_p(T) = 0$, and by the definition of a positive presentation, we see that there is an $n \geq 1$ such that $T \subseteq \sigma[n]^+$. Therefore the procedure refutes the class $\mathcal{C}$ from $\sigma[n]$ and stops.

(II) In case $L \in \mathcal{C}$. As easily seen, for any finite set $T \subseteq L$, the value of $\mathrm{econs}_p(T)$ never becomes 0. Since the IIM M infers L w.r.t. $\mathcal{C}$ in the limit from σ, it follows that the procedure infers L w.r.t. $\mathcal{C}$ in the limit from σ.

Furthermore it is easy to see that the procedure works consistently and responsively. □

By Proposition 5, 7, Theorem 8 and 9, we have the following Corollary 10.

Corollary 10. *(a) For a class $\mathcal{C}$, the following three statements are equivalent:*

(i) $\mathcal{C}$ is refutably inferable from positive data.

(ii) $\mathcal{C}$ is refutably, consistently and responsively inferable from positive data.

(iii) $\mathcal{C}$ satisfies the conditions (3.1), (3.3) and (3.5).

(b) For a class $\mathcal{C}$, the following three statements are equivalent:

(i) $\mathcal{C}$ is refutably inferable from complete data.

(ii) $\mathcal{C}$ is refutably, consistently and responsively inferable from complete data.

(iii) $\mathcal{C}$ satisfies the conditions (3.2) and (3.4).

The above conditions heavily depend on each concept rather than the properties of the class concerned. Thus we need to investigate another conditions concerned with the properties of the class itself.

Definition. A class $\mathcal{C}$ is said to be *closed under the subset operation*, if for any $L_i \in \mathcal{C}$, all nonempty subsets of L_i are also in the class $\mathcal{C}$.

A class $\mathcal{C}$ is said to be *of finite hierarchy*, if there is no infinite sequence of concepts $L_{i_1}, L_{i_2}, \cdots \in \mathcal{C}$ such that $L_{i_1} \subsetneq L_{i_2} \subsetneq \cdots$.

As easily seen, if a class $\mathcal{C}$ has the so-called finite elasticity[37], then $\mathcal{C}$ is of finite hierarchy. Moreover, if a class $\mathcal{C}$ is closed under the subset operation and it is of finite hierarchy, then $\mathcal{C}$ contains no infinite concept, as shown in the proof of Lemma 12.

Lemma 11. *If a class $\mathcal{C}$ is refutably inferable from positive data, then $\mathcal{C}$ satisfies the following two conditions:*

(3.6) $\mathcal{C}$ is closed under the subset operation.

(3.7) $\mathcal{C}$ is of finite hierarchy.

Proof. (I) Suppose that the condition (3.6) does not hold, that is, there is a nonempty concept L such that $L \notin \mathcal{C}$ and $L \subseteq L_i$ for some $L_i \in \mathcal{C}$. Then every subset of L is consistent with L_i. Therefore by Proposition 5, $\mathcal{C}$ is not refutably inferable from positive data.

(II) Assume that a class $\mathcal{C}$ is refutably inferable from positive data. Then by the above (I), the condition (3.6) holds.

Claim: The class $\mathcal{C}$ contains no infinite concept.

Proof of the claim. Suppose that $\mathcal{C}$ contains an infinite concept L_i. Then by the condition (3.6), we see that $\mathcal{C}$ contains all nonempty finite subset of L_i. We can show that any RIIM does not infer L_i w.r.t. $\mathcal{C}$ in the limit from a certain positive presentation of L_i as in Gold[12]. This is a contradiction. ■

Now suppose that the condition (3.7) does not hold, that is, there is an infinite sequence of concepts $L_{i_1}, L_{i_2}, \cdots \in \mathcal{C}$ such that $L_{i_1} \subsetneq L_{i_2} \subsetneq \cdots$. Then we consider the concept $L = \bigcup_{j=0}^{\infty} L_{i_j}$. Since L is an infinite concept, it follows by the above claim that $L \notin \mathcal{C}$. By the definition of the concept L, we see that for any finite set $T \subseteq L$, there is an $L_{i_j} \in \mathcal{C}$ such that T is consistent with L_{i_j}. This contradicts the assumption by Proposition 5. □

Lemma 12. *If a class $\mathcal{C}$ satisfies the following three conditions, then $\mathcal{C}$ is refutably, consistently and responsively inferable from positive data.*

(3.6) $\mathcal{C}$ is closed under the subset operation.

(3.7) $\mathcal{C}$ is of finite hierarchy.

(3.3) The function econs_p for $\mathcal{C}$ is recursive.

Proof. Assume that a class $\mathcal{C}$ satisfies the above three conditions. Then by Theorem 9, it suffices for us to show that $\mathcal{C}$ satisfies the conditions (3.1) and (3.5).

Claim: The class $\mathcal{C}$ contains no infinite concept.

Proof of the claim. Suppose that $\mathcal{C}$ contains an infinite concept L_i. Then by the condition (3.6), all subsets of L_i are also in $\mathcal{C}$, and it follows that there is an infinite sequence of concepts $L_{i_1}, L_{i_2}, \cdots \in \mathcal{C}$ such that $L_{i_1} \subsetneq L_{i_2} \subsetneq \cdots$, which contradicts the condition (3.7). ■

(I) By this claim, we see that $\mathcal{C}$ is inferable in the limit from positive data, that is, the condition (3.5) is satisfied (cf. Angluin[2]).

(II) Let $L \notin \mathcal{C}$ be a nonempty concept. (i) In case L is a finite concept. By the condition (3.6), we see that L is *not* consistent with any $L_i \in \mathcal{C}$, because the existence of $L_i \in \mathcal{C}$ with $L \subseteq L_i$ means $L \in \mathcal{C}$. Therefore it suffices for us to take L itself as T in the condition (3.1). (ii) In case L is an infinite concept. Suppose that the condition (3.1) does not hold. Then for any finite set $T \subseteq L$, there is an $L_i \in \mathcal{C}$ such that T is consistent with L_i. However by the condition (3.6), we see that the above T's themselves are in $\mathcal{C}$. To sum up, every finite set $T \subseteq L$ is in $\mathcal{C}$, and it follows that there is an infinite sequence $L_{i_1}, L_{i_2}, \cdots$ such that $L_{i_1} \subsetneq L_{i_2} \subsetneq \cdots$, because L is an infinite concept. This contradicts the condition (3.7). □

By Corollary 10, Lemma 11 and 12, we have the following Theorem 13.

Theorem 13. *For a class $\mathcal{C}$, the following four statements are equivalent:*

(i) $\mathcal{C}$ is refutably inferable from positive data.

(ii) $\mathcal{C}$ is refutably, consistently and responsively inferable from positive data.

(iii) $\mathcal{C}$ satisfies the conditions (3.1), (3.3) and (3.5).

(iv) $\mathcal{C}$ satisfies the conditions (3.6), (3.7) and (3.3).

Example 1. Let $\mathcal{FC}_n$ be the class of all nonempty finite concepts each of which cardinality is *just* n. Then this class is *not* refutably inferable from positive data for any $n \geq 2$, because it is not closed under the subset operation.

In contrast with the above class, let $\mathcal{FC}_{\leq n}$ be the class of all nonempty finite concepts each of which cardinality is *at most* n. We note that $\mathcal{FC}_{\leq 1} = \mathcal{FC}_1$. As easily seen, the function econs_p for $\mathcal{FC}_{\leq n}$ is recursive, because $\mathrm{econs}_p(T) = 1$ if and only if the cardinality of T is not greater than n. Furthermore, this class is closed under the subset operation and of finite hierarchy. Therefore $\mathcal{FC}_{\leq n}$ is refutably inferable from positive data.

Lastly, let $\mathcal{FC}_*$ be the class of all nonempty finite concepts (cf. Corollary 6). The function econs_p for $\mathcal{FC}_*$ is recursive, because $\mathrm{econs}_p(T) = 1$ for any finite set $T \subseteq U$. This class is closed under the subset operation but is not of finite hierarchy. Therefore $\mathcal{FC}_*$ is *not* refutably inferable from positive data.

Here we present a sufficient condition for a class to be refutably inferable from complete data, which is very strict but widely applicable as shown in Section 5 and 6.

The following Lemma 14 is basic.

Lemma 14. *Let $n \geq 1$ be an integer, let $L_1, \cdots, L_n \subseteq U$ be concepts, and let $L \subseteq U$ be a concept which differs from $L_1, \cdots, L_n$. Then for any complete presentation σ of L, there is an $m \geq 1$ such that $\sigma[m]$ is* not *consistent with any concept L_i with $1 \leq i \leq n$.*

Definition. Let $\mathcal{C} = \{L_i\}_{i \in N}$ be a class, and let $\mathcal{S}$ be a subclass of $\mathcal{C}$. A set I of indices is said to be a *cover-index set* of S, if the collection of all concepts each of which has an index in I is equal to $\mathcal{S}$, that is, $\mathcal{S} = \{L_i \in \mathcal{C} \mid i \in I\}$.

Theorem 15. *If a class $\mathcal{C}$ satisfies the following two conditions, then $\mathcal{C}$ is refutably inferable from complete data.*

(3.8) For any $w \in U$, there is a uniformly and recursively generable finite cover-index set of the subclass $\{L_i \in \mathcal{C} \mid w \in L_i\}$ of $\mathcal{C}$.

(3.9) The class $\mathcal{C}$ contains the empty concept as its member.

Proof. Assume that a class $\mathcal{C}$ satisfies the conditions (3.8) and (3.9). Then let us consider the procedure in Figure 3.

Assume that we feed a complete presentation σ of a concept L to the procedure.

(I) In case $L = \phi$. It is easy to see that the while-loop (1) never terminates and that the procedure infers the empty concept w.r.t. $\mathcal{C}$ in the limit from σ.

(II) In case $L \neq \phi$. The procedure terminates the while-loop (1) in a finite time.

(i) In case $L \in \mathcal{C}$. It is easy to see that L is in the subclass $\{L_i \in \mathcal{C} \mid w \in L_i\}$, and it follows that there is an index $j \in I$ such that $L_j = L$. Since I is a finite set, we see by Lemma 14 that the for-loop (2) is eventually executed with $j \in I$ such that $L_j = L$, and the while-loop (3) never terminates. That is, the procedure infers L w.r.t. $\mathcal{C}$ in the limit from σ.

(ii) In case $L \notin \mathcal{C}$. Since I is an explicitly given finite set, we see by Lemma 14 that the procedure refutes the class $\mathcal{C}$ from σ. □

Before proceeding to the next example, we briefly recall a pattern and a pattern language. For more details, please refer to Angluin[1, 2].

Fix a nonempty finite alphabet. A pattern is a nonnull finite string of constant and variable symbols. The pattern language $L(\pi)$ generated by a pattern π is the set of all strings obtained by substituting nonnull strings of constant symbols for the variables in π. Since two patterns that are identical except for renaming of variables generate the same pattern language, we do not distinguish one from the other. We can enumerate all patterns recursively and whether $w \in L(\pi)$ or not for any w and π is recursively decidable. Therefore we can consider the class of pattern languages as an indexed family of recursive concepts, where the pattern itself is considered to be an index.

Example 2. We consider the class $\mathcal{PAT}$ of pattern languages. As easily seen, the empty concept $L = \phi$ is not in $\mathcal{PAT}$. Furthermore, for any finite set $F \subseteq U$, there is an $L_i \in \mathcal{PAT}$ such that (ϕ, F) is consistent with L_i. In fact, let l be the length of the longest string in

```
Procedure RIIM M;
begin
    T = φ;   F = φ;   i = 1;
    read_store(T, F);
    while T = φ do begin ..................................................... (1)
        while F ⊈ L_i^c do i = i + 1;
        output i;
        read_store(T, F);
    end;
    let {w} = T;
    recursively generate a cover-index set of {L_i ∈ C | w ∈ L_i}, and set it to I;
    for each j ∈ I do ........................................................ (2)
        while (T, F) is consistent with L_j do begin ........................ (3)
            output j;
            read_store(T, F);
        end;
    refute the class C and stop;
end;

Procedure read_store(T, F);
begin
    read the next fact (w, t);
    if t ='+' then T = T ∪ {w} else F = F ∪ {w};
end.
```

Figure 3: An inference machine that can refute hypothesis spaces from complete data

F. Then (ϕ, F) is consistent with the language of the pattern $x_1 x_2 \cdots x_{l+1}$. Therefore by Proposition 5, we see that $\mathcal{PAT}$ is *not* refutably inferable from complete data.

However the class $\mathcal{PAT}$ satisfies the condition (3.8). In fact, fix an arbitrary constant string w. As easily seen, if $w \in L(\pi)$, then π is not longer than w. The set of all patterns shorter than a fixed length is a recursively generable finite set, and whether $w \in L(\pi)$ or not for any w and π is recursively decidable. Therefore the set $\{\pi \mid w \in L(\pi)\}$ is a recursively generable finite set.

Thus, by Theorem 15, we see that if we add the empty concept to the class of pattern languages, then the obtained class is refutably inferable from complete data.

4. Comparisons with Other Identifications

In this section, by some distinctive examples of classes, we compare the criterion of refutable inference with some other criteria. This is motivated by the following question: What should we do if we face with facts that are inconsistent with a finitely inferred hypothesis?

In what follows, for the criterion of finite identification for a class of recursive concepts, please refer to Mukouchi[21], Lange&Zeugmann[16] and Kapur[13]. For the purpose of comparing the inferability from positive data with the inferability from complete data, we assume that all concepts are *nonempty* throughout this section.

The following Proposition 16 is obvious, because for a complete presentation σ of a *nonempty* concept L, we can effectively obtain a positive presentation of L by getting rid of all negative facts of σ and repeating a positive fact.

Proposition 16. *If a class $\mathcal{C}$ is* inferable in the limit *from* positive data, *then $\mathcal{C}$ is also* inferable in the limit *from* complete data.

Furthermore the above assertion is still valid, if we replace the phrase 'inferable in the limit' with the phrase 'finitely inferable', 'reliably inferable' or 'refutably inferable'.

Theorem 17 (Lange&Zeugmann[15]). *If a class $\mathcal{C}$ is* finitely inferable *from* complete data, *then $\mathcal{C}$ is also* inferable in the limit *from* positive data.

In the following examples, we assume appropriate universal sets and indexings of the classes.

Example 3. Again we consider the classes $\mathcal{FC}_n$, $\mathcal{FC}_{\leq n}$ and $\mathcal{FC}_*$ defined in Example 1 (cf. Theorem 3).

(A.1) For any $n \geq 2$, the class $\mathcal{FC}_n$ is *not* refutably inferable from *complete data.*

In fact, let $L \subseteq U$ be a concept with cardinality 1. It is easy to see that there are no finite sets $T \subseteq L$ and $F \subseteq L^c$ that satisfy the condition (3.2) in Proposition 5.

(A.2) For any $n \geq 1$, the class $\mathcal{FC}_n$ is reliably inferable from *positive data.*

(A.3) For any $n \geq 1$, the class $\mathcal{FC}_n$ is finitely inferable from *positive data.*

(B.1) For any $n \geq 1$, the class $\mathcal{FC}_{\leq n}$ is refutably inferable from *positive data.*

(B.2) For any $n \geq 1$, the class $\mathcal{FC}_{\leq n}$ is reliably inferable from *positive data.*

(B.3) For any $n \geq 2$, the class $\mathcal{FC}_{\leq n}$ is *not* finitely inferable from *complete data.*

(C.1) $\mathcal{FC}_*$ is *not* refutably inferable from *complete data* (cf. Corollary 6).

(C.2) $\mathcal{FC}_*$ is reliably inferable from *positive data.*

(C.3) $\mathcal{FC}_*$ is *not* finitely inferable from *complete data.*

Example 4. Let $\Sigma = \{a\}$, $L_1 = \{a^j \mid j \geq 1\}$, and $L_i = \{a^j \mid 1 \leq j \leq i-1\}$ for $i \geq 2$. Then let $\mathcal{CLC} = \{L_i\}_{i \in N}$ be the class of interest.

(D.1) The class $\mathcal{CLC}$ is refutably inferable from *complete data.*

In fact, let $L \notin \mathcal{CLC}$ be a nonempty concept. Then as easily seen, there is a $j \geq 1$ such that $a^j \notin L$ but $a^{j+1} \in L$. Therefore $T = \{a^{j+1}\}$ and $F = \{a^j\}$ satisfy the condition (3.2). It is easy to see that the condition (3.4) is also satisfied (cf. Theorem 9).

(D.2) The class $\mathcal{CLC}$ is *not* inferable from *positive data.*

We note that, in Lange&Zeugmann[17], this class was shown to be inferable within one mind change from complete data but not inferable in the limit from positive data.

Example 5. Let $\mathcal{PR}_{\leq n}$ be the class of concepts each of which consists of all multiples of at most n prime numbers. For example, the concept $\{2, 4, 6, \cdots, 7, 14, 21, \cdots\}$ is in $\mathcal{PR}_{\leq i}$ with $i \geq 2$.

(E.1) For any $n \geq 1$, the class $\mathcal{PR}_{\leq n}$ is refutably inferable from *complete data.*

In fact, let $L \in \mathcal{PR}_{\leq n}$ be a nonempty concept. (i) In case L contains more than n prime numbers. Let T be a set of some $n+1$ prime numbers in L, and let $F = \phi$. (ii) In case L contains no prime number. Let m be the least integer in L. Then let $T = \{m\}$ and $F = \{1, \cdots, m-1\}$. (iii) Otherwise. Let $p_1, \cdots, p_k$ be all prime numbers in L. As easily seen, the following (1) or (2) holds. (1) There is an $m \in L$ which is not a multiple of any p_i with $1 \leq i \leq k$. Then let $T = \{m\}$, and let F be the finite set of all prime numbers less than m each of which differs from $p_1, \cdots, p_k$. (2) There is an $m \notin L$ which is a multiple of some p_i. Then let $T = \{p_1, \cdots, p_k\}$ and $F = \{m\}$. It is easy to see that the above defined T and F satisfy the condition (3.2). Furthermore it is also easy to see that the condition (3.4) is satisfied.

(E.2) For any $n \geq 1$, the class $\mathcal{PR}_{\leq n}$ is *not* reliably inferable from *positive data.*

(E.3) For any $n \geq 2$, the class $\mathcal{PR}_{\leq n}$ is *not* finitely inferable from *complete data.*

(E.4) For any $n \geq 1$, the class $\mathcal{PR}_{\leq n}$ is inferable in the limit from *positive data.*

Let $\mathcal{PR}_n$ be the class of concepts each of which consists of all multiples of n distinct prime numbers. For example, the concept $\{2, 4, 6, \cdots, 7, 14, 21, \cdots\}$ is in $\mathcal{PR}_2$, but not in $\mathcal{PR}_i$ with $i = 1$ or $i \geq 3$. We note that $\mathcal{PR}_1 = \mathcal{PR}_{\leq 1}$.

(F.1) For any $n \geq 2$, the class $\mathcal{PR}_n$ is *not* refutably inferable from *complete data.*

In fact, let $L = \{2\}$. Then for any finite set $F \subseteq L^c$, there is a prime number which is greater than any integer in F. Therefore there are no finite sets $T \subseteq L$ and $F \subseteq L^c$ that satisfy the condition (3.2).

(F.2) For any $n \geq 1$, the class $\mathcal{PR}_n$ is *not* reliably inferable from *positive data.*

(F.3) For any $n \geq 1$, the class $\mathcal{PR}_n$ is finitely inferable from *positive data.*

Example 6. Again we consider the class $\mathcal{PAT}$ of pattern languages. As easily seen from Theorem 15 and Example 2, the empty concept is only the concept that does not satisfy the condition (3.2). Since all concepts are assumed to be nonempty in this section, the class of pattern languages is shown to be refutably inferable from complete data.

(G.1) $\mathcal{PAT}$ is refutably inferable from *complete data.*

(G.2) $\mathcal{PAT}$ is *not* reliably inferable from *positive data.*

(G.3) $\mathcal{PAT}$ is finitely inferable from *complete data* but *not* finitely inferable from *positive data* (cf. Mukouchi[21], Lange&Zeugmann[15]).

(G.4) $\mathcal{PAT}$ is inferable in the limit from *positive data* (cf. Angluin[2]).

We can summarize the above comparisons in Figure 4.

In the figure, the prefix 'LIM', 'FIN', 'REL' or 'REF' means the collection of all classes that are 'inferable in the limit', 'finitely inferable', 'reliably inferable' or 'refutably inferable', respectively, and the postfix 'TXT' or 'INF' means 'from positive data' or 'from complete data', respectively. For example, REF-TXT is the collection of all classes that are refutably inferable from positive data.

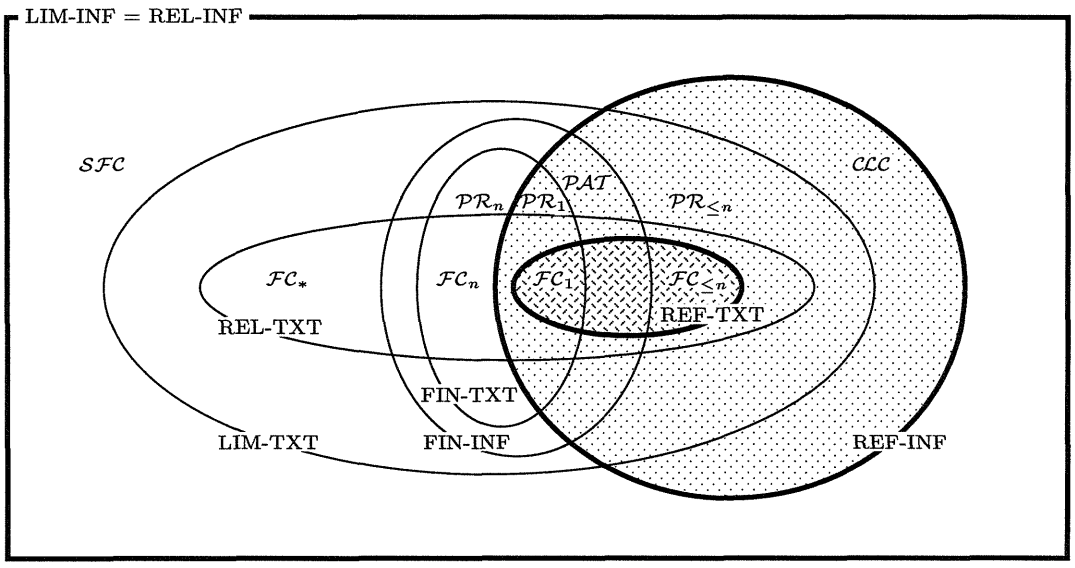

Figure 4: Comparisons with other identifications

The classes $\mathcal{FC}_1$, $\mathcal{FC}_n$, $\mathcal{FC}_{\leq n}$ and $\mathcal{FC}_*$ consist of all finite concepts each of which cardinality is just 1, $n \geq 2$, at most n and unrestricted finite, respectively (cf. Example 3). The class $\mathcal{CLC}$ has been shown in Example 4. The classes $\mathcal{PR}_1$, $\mathcal{PR}_n$ and $\mathcal{PR}_{\leq n}$ consist of all multiples of a prime number, $n \geq 2$ distinct prime numbers and at most n prime numbers, respectively (cf. Example 5). The class $\mathcal{PAT}$ is the class of pattern languages (cf. Example 6). The class $\mathcal{SFC}$ is the so-called superfinite class[12], that is, a class contains all finite concepts and at least one infinite concept.

In Figure 4, we see that a subclass of a refutably inferable class is not always refutably inferable.

5. Unions of Some Classes

In this section we consider two types of union classes. First we take a class as the collection of all concepts from n classes.

Definition. Let $n \geq 1$ be an integer, and let $\mathcal{C}_1, \cdots, \mathcal{C}_n$ be classes. For i with $1 \leq i \leq n$ and $j \geq 1$, the j-th concept L_j of the class $\mathcal{C}_i$ is denoted by $L_{(i,j)}$. Then the union class of $\mathcal{C}_1, \cdots, \mathcal{C}_n$ is represented as:

$$\bigcup_{i=1}^{n} \mathcal{C}_i = \{L_{(i,j)}\}_{1 \leq i \leq n,\, j \in N}.$$

By assuming a bijective coding from $\{1, \cdots, n\} \times N$ to N, the new class above becomes an indexed family of recursive concepts.

Theorem 18. *Let $n \geq 1$ be an integer, and let $\mathcal{C}_1, \cdots, \mathcal{C}_n$ be classes each of which is refutably inferable from positive data (resp., complete data). Then the class $\bigcup_{i=1}^{n} \mathcal{C}_i$ is refutably inferable from positive data (resp., complete data).*

Proof. We only give the proof of the case of positive data. The proof for complete data can be given in a similar way.

For any i with $1 \leq i \leq n$, let M_i be an RIIM which refutably infers $\mathcal{C}_i$ from positive data. Then let us consider the procedure in Figure 5.

Procedure RIIM M;
begin
  for $i = 1$ to n **do**
    while M_i does not refute the class $\mathcal{C}_i$ **do begin**
      simulate M_i with presenting facts read so far;
      during the simulation,
        if M_i requests another fact **then**
          read the next fact and present it to M_i;
        if M_i produces a guess j **then**
          output the coding of (i, j);
    end;
  refute the class $\bigcup_{i=1}^{n} \mathcal{C}_i$ and stop;
end.

Figure 5: An RIIM for unions by $\bigcup$

Assume that we feed a positive presentation σ of a nonempty concept L to the procedure.

(I) In case $L \notin \bigcup_{i=1}^{n} \mathcal{C}_i$. Then for any i with $1 \leq i \leq n$, $L \notin \mathcal{C}_i$ holds, and it follows that M_i refutes the class $\mathcal{C}_i$ from σ. Thus the procedure refutes the class $\bigcup_{i=1}^{n} \mathcal{C}_i$ from σ.

(II) In case $L \in \bigcup_{i=1}^{n} \mathcal{C}_i$. Let i_0 be the least integer such that $L \in \mathcal{C}_{i_0}$. Then for any i with $1 \leq i < i_0$, $L \notin \mathcal{C}_i$ holds, and it follows that M_i refutes the class $\mathcal{C}_i$ from σ. Therefore the for-loop in the procedure reaches the case of $i = i_0$. Since M_{i_0} infers L w.r.t. $\mathcal{C}_{i_0}$ in the limit from σ, it follows that M_{i_0} converges to an index j with $L_{(i_0,j)} = L$ for σ. Thus the procedure converges to the coding of (i_0, j) for σ. That is, the procedure infers L w.r.t. $\bigcup_{i=1}^{n} \mathcal{C}_i$ in the limit from σ.

Thus the procedure refutably infers the class $\bigcup_{i=1}^{n} \mathcal{C}_i$ from positive data. □

Now we consider a class of concepts each of which is a union of at most n concepts from n classes (cf. Wright[37]).

Definition. Let $n \geq 1$ be an integer, and let $\mathcal{C}_1, \cdots, \mathcal{C}_n$ be classes. For i with $1 \leq i \leq n$ and $j \geq 0$, $L_{\langle i,j \rangle}$ denotes the empty concept if $j = 0$, otherwise the j-th concept L_j of the class $\mathcal{C}_i$. Then we define a class generated by $\mathcal{C}_1, \cdots, \mathcal{C}_n$ as follows:

$$\bigsqcup_{i=1}^{n} \mathcal{C}_i = \left\{ \bigcup_{i=1}^{n} L_{\langle i, j_i \rangle} \;\middle|\; (j_1, \cdots, j_n) \in \mathcal{N}^n \right\},$$

where $\mathcal{N}^n$ is the set of all n-tuples of nonnegative integers, that is, $\mathcal{N}^n = \{0, 1, 2, \cdots\}^n$.

By assuming a bijective coding from $\mathcal{N}^n$ to N, the new class above becomes an indexed family of recursive concepts.

If each class satisfies the condition (3.8), then the above class is shown to be refutably inferable from complete data.

Theorem 19. *Let $n \geq 1$ be an integer, and let $\mathcal{C}_1, \cdots, \mathcal{C}_n$ be classes each of which satisfies the condition (3.8). Then the class $\bigsqcup_{i=1}^n \mathcal{C}_i$ is refutably inferable from complete data.*

Proof. Let us consider the procedure in Figure 6, where the procedure read_store is the same one as in Figure 3 and λ is a special element not in the universal set U.

Assume that we feed a complete presentation σ of a concept L_{base} to the procedure.

(A) In case $L_{\text{base}} \in \bigsqcup_{i=1}^n \mathcal{C}_i$. Let $NE(m) = \{(j_1, \cdots, j_n) \in \mathcal{N}^n \mid$ there are just m nonempty concepts among $L_{\langle 1,j_1\rangle}, \cdots, L_{\langle n,j_n\rangle}\}$.

Claim: In the procedure, for any m with $0 \leq m \leq n$, if T_m and F_m are defined, then (T_m, F_m) is *not* consistent with $(L_{\langle 1,j_1\rangle} \cup \cdots \cup L_{\langle n,j_n\rangle})$ for any $(j_1, \cdots, j_n) \in NE(m)$.

Proof of the claim. This proof is given by mathematical induction on m.

(I) In case $m = 0$. It is clear because T_0 is nonempty and the union of n empty concepts is empty.

(II) In case $m \geq 1$. We assume the claim for $m-1$, and assume that T_m and F_m are defined. Then suppose that there is an n-tuple $(k_1, \cdots, k_n) \in NE(m)$ such that (T_m, F_m) is consistent with $(L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})$. Then (T_{m-1}, F_{m-1}) is consistent with $(L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})$, because $T_{m-1} \subseteq T_m$ and $F_{m-1} \subseteq F_m$ hold.

Here suppose that there is an $i \geq 1$ such that $L_{\langle i,k_i\rangle} \neq \phi$ and $(L_{\langle i,k_i\rangle} \cap T_{m-1}) = \phi$. Then $T_{m-1} \subseteq (L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle i-1,k_{i-1}\rangle} \cup L_{\langle i+1,k_{i+1}\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})$ and $F_{m-1} \subseteq (L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})^c \subseteq (L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle i-1,k_{i-1}\rangle} \cup L_{\langle i+1,k_{i+1}\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})^c$ hold. This means that (T_{m-1}, F_{m-1}) is consistent with $(L_{\langle 1,k_1\rangle} \cup \cdots \cup L_{\langle i-1,k_{i-1}\rangle} \cup L_{\langle i+1,k_{i+1}\rangle} \cup \cdots \cup L_{\langle n,k_n\rangle})$. This contradicts the induction hypothesis, because $(k_1, \cdots, k_{i-1}, 0, k_{i+1}, \cdots, k_n) \in NE(m-1)$.

Thus we have $(L_{\langle i,k_i\rangle} \cap T_{m-1}) \neq \phi$ for any i with $L_{\langle i,k_i\rangle} \neq \phi$. Therefore we can take u_i's as follows: If $L_{\langle i,k_i\rangle} \neq \phi$, then $u_i \in (L_{\langle i,k_i\rangle} \cap T_{m-1})$, otherwise $u_i = \lambda$. Since all u_i's are in $(T_{m-1} \cup \{\lambda\})$ and the number of u_i's other than λ is just m, the n-tuple $(u_1, \cdots, u_n)$ is in $\mathcal{T}_m$. Thus there is a case where the for-loop (2) is executed with $(u_1, \cdots, u_n)$. In this case there is an n-tuple $(k'_1, \cdots, k'_n) \in S_1 \times \cdots \times S_n$ such that for any i with $1 \leq i \leq n$, $L_{k'_i} = L_{k_i}$. Thus there is a case where the for-loop (3) is executed with $(k'_1, \cdots, k'_n)$. Since (T_m, F_m) is consistent with $(L_{\langle 1,k'_1\rangle} \cup \cdots \cup L_{\langle n,k'_n\rangle})$, it follows that for any finite sets $T \subseteq T_m$ and $F \subseteq F_m$, (T, F) is consistent with $(L_{\langle 1,k'_1\rangle} \cup \cdots \cup L_{\langle n,k'_n\rangle})$. However the while-loop (4) should terminate with $T \subseteq T_m$ and $F \subseteq F_m$, because T_m and F_m are assumed to be defined. This is a contradiction. ■

Since $L_{\text{base}} \in \bigsqcup_{i=1}^n \mathcal{C}_i$, there are an m_0 with $0 \leq m_0 \leq n$ and an n-tuple $(j_1, \cdots, j_n)$ such that $(j_1, \cdots, j_n) \in NE(m_0)$ and $L_{\text{base}} = (L_{\langle 1,j_1\rangle} \cup \cdots \cup L_{\langle n,j_n\rangle})$. By this claim, we see that T_{m_0} and F_{m_0} are never defined. By Lemma 14, this means that the procedure outputs the coding of an n-tuple $(j_1, \cdots, j_n)$ with $L_{\text{base}} = (L_{\langle 1,j_1\rangle} \cup \cdots \cup L_{\langle n,j_n\rangle})$ and never terminates the while-loop (1) or (4).

(B) In case $L_{\text{base}} \notin \bigsqcup_{i=1}^n \mathcal{C}_i$. By using Lemma 14 n times, we see that the procedure refutes the class $\bigsqcup_{i=1}^n \mathcal{C}_i$ from σ. □

Procedure RIIM M;
begin
  $T = \phi$; $F = \phi$;
  read_store(T, F);
  while $T = \phi$ **do begin** ..... (1)
    output the coding of $(0, \cdots, 0)$;
    read_store(T, F);
  end;
  $T_0 = T$; $F_0 = F$;
  for $m = 1$ to n **do begin**
    let $\mathcal{T}_m = \left\{ \begin{array}{c|c} (w_1, \cdots, w_n) & \text{the number of } w_i\text{'s} \\ \in (T_{m-1} \cup \{\lambda\})^n & \text{other than } \lambda \text{ is just } m \end{array} \right\}$;

    for each $(w_1, \cdots, w_n) \in \mathcal{T}_m$ **do begin** ..... (2)
      for $i = 1$ to n **do**
        if $w_i = \lambda$ **then** let $S_i = \{0\}$
        else recursively generate a cover-index set w.r.t. $\mathcal{C}_i$
          of $\{L_{\langle i,j \rangle} \in \mathcal{C}_i \mid w_i \in L_{\langle i,j \rangle}\}$, and set it to S_i;
      if all S_i's are nonempty **then**
        for each $(j_1, \cdots, j_n) \in S_1 \times \cdots \times S_n$ **do** ..... (3)
          while (T, F) is consistent with $(L_{\langle 1,j_1 \rangle} \cup \cdots \cup L_{\langle n,j_n \rangle})$ **do begin** . (4)
            output the coding of $(j_1, \cdots, j_n)$;
            read_store(T, F);
          end;
    end;
    $T_m = T$; $F_m = F$;
  end;
  refute the class $\bigsqcup_{i=1}^{n} \mathcal{C}_i$ and stop;
end.

Figure 6: An RIIM for unions by $\bigsqcup$

Example 7. We consider the class $\mathcal{PAT}$ of pattern languages. As shown in Example 2, the class satisfies the condition (3.8). Therefore by Theorem 19, for any $n \geq 1$, the class of unions of at most n pattern languages is refutably inferable from complete data.

By Corollary 6, we see that if the number of patterns is not bounded by a constant number, then the class is not refutably inferable from complete data, because it contains all nonempty finite languages.

We note that for any $n \geq 1$, the class of unions of at most n pattern languages is shown to be inferable in the limit from positive data (cf. Wright[37], Shinohara[30, 31]).

6. EFS Definable Classes

In this section, we consider so-called model inference (cf. Shapiro[27]) and language learning using *elementary formal systems* (*EFS*'s, for short).

The EFS's were originally introduced by Smullyan[34] to develop his recursion theory. In a word, EFS's are a kind of logic programming language which uses patterns instead of terms in first order logic[38], and they are shown to be natural devices to define languages[5].

In this paper, we briefly recall EFS's. For detailed definitions and properties of EFS's, please refer to Smullyan[34], Arikawa[5], Arikawa et al.[6, 7] and Yamamoto[38].

Let Σ, X and Π be mutually disjoint nonempty sets. We assume that Σ is finite, and fix it throughout this section. Elements in Σ, X and Π are called constant symbols, variables and predicate symbols, respectively. By $p, q, p_1, p_2, \cdots$, we denote predicate symbols. Each predicate symbol is associated with a positive integer which we call an *arity*.

In general, for a set S, S^+ denotes the set of all nonempty finite strings over S, and $\sharp S$ denotes the cardinality of S.

Definition. A *term*, or a *pattern*, is an element in $(\Sigma \cup X)^+$, that is, it is a nonnull string over $(\Sigma \cup X)$. By $\pi, \pi_1, \pi_2, \cdots$, we denote terms. A term π is said to be *ground*, if $\pi \in \Sigma^+$. By $w, w_1, w_2, \cdots$, we denote ground terms.

An *atomic formula* (*atom*, for short) is an expression of the form $p(\pi_1, \cdots, \pi_n)$, where p is a predicate symbol with arity n, and $\pi_1, \cdots, \pi_n$ are terms. By $A, B, A_1, A_2, \cdots$, we denote atoms. An atom $p(\pi_1, \cdots, \pi_n)$ is said to be ground, if $\pi_1, \cdots, \pi_n$ are ground terms.

We define well-formed formulas and clauses in the ordinary ways[18].

Definition. A *definite clause* is a clause of the form

$$A \leftarrow B_1, \cdots, B_n,$$

where $n \geq 0$, and $A, B_1, \cdots, B_n$ are atoms. The atom A above is called the *head* of the clause, and the sequence $B_1, \cdots, B_n$ is called the *body* of the clause. By $C, D, C_1, C_2, \cdots$, we denote definite clauses. Then an *EFS* is a finite set of definite clauses, each of which is called an *axiom*.

A *substitution* is a homomorphism from terms to terms which maps each symbol $a \in \Sigma$ to itself.

In the world of EFS's, the *Herbrand base* (*HB*, for short) is the set of all ground atoms. A subset I of HB is called an *Herbrand interpretation.* We also define *Herbrand model*, and *the least Herbrand model* in the ordinary ways[18].

For an EFS Γ, the least Herbrand model is denoted by $M(\Gamma)$. For an EFS Γ and a predicate symbol p with arity n, we define the set of n-tuples of ground terms as follows:

$$L(\Gamma, p) = \{(w_1, \cdots, w_n) \in (\Sigma^+)^n \mid p(w_1, \cdots, w_n) \in M(\Gamma)\}.$$

In case the arity of p is 1, i.e. p is unary, we regard $L(\Gamma, p)$ as a language over Σ.

Now we put a syntactical restriction on EFS's, because the least Herbrand model $M(\Gamma)$ for an unrestricted EFS Γ may not be recursive, that is, for a ground atom A, we can not recursively decide whether $A \in M(\Gamma)$ or not.

For a term π, $\|\pi\|$ denotes the length of π, and $o(x, \pi)$ denotes the number of all occurrences of a variable x in π. For an atom $p(\pi_1, \cdots, \pi_n)$, we define the length of the atom and the number of variable's occurrences in the atom as follows:

$$\begin{aligned} \|p(\pi_1, \cdots, \pi_n)\| &= \|\pi_1\| + \cdots + \|\pi_n\|, \\ o(x, p(\pi_1, \cdots, \pi_n)) &= o(x, \pi_1) + \cdots + o(x, \pi_n). \end{aligned}$$

Definition. A clause $A \leftarrow B_1, \cdots, B_n$ is said to be *length-bounded*, if

$$\|A\theta\| \geq \|B_1\theta\| + \cdots + \|B_n\theta\|$$

for any substitution θ.

An EFS Γ is said to be length-bounded, if all axioms of Γ are length-bounded.

The notion of length-bounded clauses is characterized by the following Lemma 20.

Lemma 20 (Arikawa et al.[6, 7]). *A clause $A \leftarrow B_1, \cdots, B_n$ is length-bounded, if and only if $\|A\| \geq \|B_1\| + \cdots + \|B_n\|$ and $o(x, A) \geq o(x, B_1) + \cdots + o(x, B_n)$ hold for any variable x.*

From now on, we only consider length-bounded EFS's. For length-bounded EFS's, the following Theorem 21 holds.

Theorem 21 (Arikawa et al.[6, 7], Yamamoto[38]). *For a length-bounded EFS Γ, the least Herbrand model $M(\Gamma)$ is recursive, that is, for any ground atom A, whether $A \in M(\Gamma)$ or not is recursively decidable.*

Furthermore the following Theorem 22 shows the power of length-bounded EFS's.

Theorem 22 (Arikawa et al.[6, 7]). *A language $L \subseteq \Sigma^+$ is context-sensitive, if and only if L is definable by a length-bounded EFS.*

We devote the rest of this section to investigating the refutable inferability of length-bounded EFS definable classes. We adapt the definitions of inferability to the case of EFS's as follows, but as easily seen, the essential part is kept unchanged from definitions in Section2. In what follows, we assume that outputs from an IIM or an RIIM are EFS's.

Definition. For an atom A, pred(A) denotes the predicate symbol of A. For a set $\Pi_0 \subseteq \Pi$ and a set S of atoms, $S|_{\Pi_0}$ denotes the set of atoms restricted to Π_0, that is,

$$S|_{\Pi_0} = \{A \in S \mid \text{pred}(A) \in \Pi_0\}.$$

A *predicate-restricted complete presentation* of a set $I \subseteq HB$ w.r.t. $\Pi_0 \subseteq \Pi$ is an infinite sequence $(A_1, t_1), (A_2, t_2), \cdots$ of elements in $HB|_{\Pi_0} \times \{+, -\}$ such that $\{A_i \mid t_i = +, i \geq 1\} = I|_{\Pi_0}$ and $\{A_i \mid t_i = -, i \geq 1\} = HB|_{\Pi_0} \setminus I|_{\Pi_0}$.

An IIM M or an RIIM M is said to *converge to an EFS Γ for a presentation σ*, if there is an $n \geq 1$ such that for any $m \geq n$, $\overline{M}(\sigma[m]) = \Gamma$.

Let $\mathcal{EC}$ be a class of EFS's. For an EFS $\Gamma \in \mathcal{EC}$ and a predicate-restricted complete presentation σ of $M(\Gamma)$ w.r.t. $\Pi_0 \subseteq \Pi$, an IIM M or an RIIM M is said to *infer the EFS Γ w.r.t. $\mathcal{EC}$ in the limit from σ*, if M converges to an EFS $\Gamma' \in \mathcal{EC}$ with $M(\Gamma')|_{\Pi_0} = M(\Gamma)|_{\Pi_0}$ for σ.

A class $\mathcal{EC}$ is said to be *theoretical-term-freely inferable in the limit from complete data*, if for any nonempty finite subset Π_0 of Π, there is an IIM M which infers Γ w.r.t. $\mathcal{EC}$ in the limit from σ for any EFS $\Gamma \in \mathcal{EC}$ and any predicate-restricted complete presentation σ of $M(\Gamma)$ w.r.t. Π_0.

A class $\mathcal{EC}$ is said to be *theoretical-term-freely and refutably inferable from complete data*, if for any nonempty finite subset Π_0 of Π, there is an RIIM M which satisfies the following condition: For any set $I \subseteq HB$ and any predicate-restricted complete presentation σ of I w.r.t. Π_0, (i) if there is an EFS $\Gamma \in \mathcal{EC}$ such that $M(\Gamma)|_{\Pi_0} = I|_{\Pi_0}$, then M infers Γ w.r.t. $\mathcal{EC}$ in the limit from σ, (ii) otherwise M refutes the class $\mathcal{EC}$ from σ.

Theoretical terms are supplementary predicates that are necessary for defining some goal predicates. In the above definition, the phrase 'theoretical-term-freely inferable' means that *from only the facts on the goal predicates*, an IIM or an RIIM generates some suitable predicates and infers an EFS which explains the goal predicates.

Definition. For two EFS's Γ and Γ', and a set $\Pi_0 \subseteq \Pi$, we write $\Gamma \equiv_{\Pi_0} \Gamma'$, if we can make Γ' identical to Γ by renaming predicate symbols other than those in Π_0 and by renaming variables in each axiom. We assume some canonical form of an EFS w.r.t. Π_0, and canon(Γ, Π_0) denotes the representative EFS for the set of EFS's $\{\Gamma' \mid \Gamma \equiv_{\Pi_0} \Gamma'\}$.

For an EFS Γ, $HPRED(\Gamma)$ (resp., $BPRED(\Gamma)$) denotes the set of all predicate symbols appearing in the heads (resp., the bodies) of the axioms of Γ, and $PRED(\Gamma)$ denotes the set $HPRED(\Gamma) \cup BPRED(\Gamma)$. We also define various sets and classes as follows:

$$\begin{aligned}
\mathcal{LB}^{[n]} &= \{\Gamma \mid \Gamma \text{ is a length-bounded EFS with } n \text{ axioms}\}, \\
\mathcal{LB}^{[n]}[\Pi_0] &= \{\text{canon}(\Gamma, \Pi_0) \mid \Gamma \in \mathcal{LB}^{[n]}\}, \\
\mathcal{MLB}^{[n]}[\Pi_0] &= \{\Gamma \in \mathcal{LB}^{[n]}[\Pi_0] \mid BPRED(\Gamma) \subseteq HPRED(\Gamma)\}, \\
\mathcal{MLB}^{[n]}[\Pi_0](l) &= \left\{\Gamma \in \mathcal{MLB}^{[n]}[\Pi_0] \;\middle|\; \begin{array}{c}\text{the head's length of each axiom} \\ \text{of } \Gamma \text{ is not greater than } l.\end{array}\right\},
\end{aligned}$$

$$\mathcal{LB}^{[\leq n]} = \bigcup_{i=0}^{n} \mathcal{LB}^{[i]}, \qquad \mathcal{MLB}^{[\leq n]}[\Pi_0] = \bigcup_{i=0}^{n} \mathcal{MLB}^{[i]}[\Pi_0],$$

$$\mathcal{MLB}^{[\leq n]}[\Pi_0](l) = \bigcup_{i=0}^{n} \mathcal{MLB}^{[i]}[\Pi_0](l),$$

where $l, n \geq 0$, and $\Pi_0 \subseteq \Pi$.

We note that for any EFS's Γ and Γ', and any $\Pi_0 \subseteq \Pi$, whether $\Gamma \equiv_{\Pi_0} \Gamma'$ or not is recursively decidable, and that we can effectively obtain the EFS $\mathrm{canon}(\Gamma, \Pi_0)$.

We prepare some basic lemmas.

Lemma 23 (Shinohara[31]). *Let Γ be a length-bounded EFS, and let $A \in M(\Gamma)$ be a ground atom. Then if Γ has an axiom C whose head is longer than A, then $A \in M(\Gamma \setminus \{C\})$ holds.*

Lemma 24. *For any EFS $\Gamma \in \mathcal{LB}^{[\leq n]}$ and any $\Pi_0 \subseteq \Pi$, there is an EFS $\Gamma' \in \mathcal{MLB}^{[\leq n]}[\Pi_0]$ such that $M(\Gamma')|_{\Pi_0} = M(\Gamma)|_{\Pi_0}$.*

Proof. Let $\Gamma \in \mathcal{LB}^{[\leq n]}$ and $\Gamma_1 = \mathrm{canon}(\Gamma, \Pi_0)$. Then $M(\Gamma_1)|_{\Pi_0} = M(\Gamma)|_{\Pi_0}$ holds. Therefore if $BPRED(\Gamma_1) \subseteq HPRED(\Gamma_1)$, then Γ_1 itself is in $\mathcal{MLB}^{[\leq n]}[\Pi_0]$. Otherwise, let Γ_2 be the EFS which is obtained from Γ_1 by subtracting all axioms that have predicate symbols in $BPRED(\Gamma_1) \setminus HPRED(\Gamma_1)$ in their bodies, and let $\Gamma_3 = \mathrm{canon}(\Gamma_2, \Pi_0)$. Then clearly, $M(\Gamma_3)|_{\Pi_0} = M(\Gamma)|_{\Pi_0}$ and $\Gamma_3 \in \mathcal{MLB}^{[\leq n]}[\Pi_0]$ hold. □

Lemma 25. *Let $n \geq 1$ be an integer, let $\Pi_0 \subseteq \Pi$ be a set of predicates, let $T \subseteq HB|_{\Pi_0}$ be a nonempty finite set, and let $F \subseteq HB|_{\Pi_0}$ be a finite set. Assume that (T, F) is* not *consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[n-1]}[\Pi_0]$. Then for any EFS $\Gamma \in \mathcal{MLB}^{[n]}[\Pi_0] \setminus \mathcal{MLB}^{[n]}[\Pi_0](l)$, (T, F) is* not *consistent with $M(\Gamma)|_{\Pi_0}$, where $l = \max\{\|A\| \mid A \in T\}$.*

Proof. Let $\Gamma \in \mathcal{MLB}^{[n]}[\Pi_0] \setminus \mathcal{MLB}^{[n]}[\Pi_0](l)$. Then there is a length-bounded clause $C \in \Gamma$ whose head's length is greater than l. Let $\Gamma_1 = \Gamma \setminus \{C\}$. Suppose that (T, F) is consistent with $M(\Gamma)|_{\Pi_0}$. Since the head of C is longer than every ground atom in T, it follows by Lemma 23 that $T \subseteq M(\Gamma_1)|_{\Pi_0}$. Furthermore, since $\Gamma_1 \subseteq \Gamma$, it follows that $M(\Gamma_1) \subseteq M(\Gamma)$ by monotonicity of the least Herbrand model. Therefore $F \subseteq HB|_{\Pi_0} \setminus M(\Gamma)|_{\Pi_0} \subseteq HB|_{\Pi_0} \setminus M(\Gamma_1)|_{\Pi_0}$ holds. Thus (T, F) is consistent with $M(\Gamma_1)|_{\Pi_0}$. Let $\Gamma_2 = \mathrm{canon}(\Gamma_1, \Pi_0)$. Then $M(\Gamma_2)|_{\Pi_0} = M(\Gamma_1)|_{\Pi_0}$ holds, and it follows that (T, F) is consistent with $M(\Gamma_2)|_{\Pi_0}$. This contradicts the assumption, because $\Gamma_2 \in \mathcal{MLB}^{[n-1]}[\Pi_0]$. □

Lemma 26. *For any $l, n \geq 0$ and any finite subset Π_0 of Π, the set $\mathcal{MLB}^{[n]}[\Pi_0](l)$ is a uniformly and recursively generable finite set.*

Proof. As easily seen, for any EFS $\Gamma \in \mathcal{MLB}^{[n]}[\Pi_0](l)$, the number of predicate symbols appearing in Γ is at most n and the arities of those predicate symbols are at most l.

Put $PA(l, m) = \{\{q_1^{(j_1)}, \cdots, q_k^{(j_k)}\} \mid 0 \leq k \leq m \text{ and } 1 \leq j_1 \leq \cdots \leq j_k \leq l\}$, where for a predicate symbol $q_i^{(t)}$, the superscript t represents its arity. Then put $PR(l, n, \Pi_0) = \{\Pi' \cup \Pi'' \mid \Pi' \subseteq \Pi_0 \text{ and } \Pi'' \in PA(l, n - \sharp\Pi')\}$. By the above observation, it is sufficient for us to generate EFS's Γ with $PRED(\Gamma) \in PR(l, n, \Pi_0)$, because we do not distinguish two EFS's that are identical except for renaming of predicate symbols other than those in Π_0.

As easily seen, the above $PR(l, n, \Pi_0)$ is a uniformly and recursively generable finite set. Furthermore the set of all terms shorter than a fixed length is a uniformly and recursively generable finite set, where we do not distinguish two terms that are identical except for renaming of variables.

Roughly speaking, we recursively generate EFS's in $\mathcal{MLB}^{[n]}[\Pi_0](l)$ as follows: We combine sets of predicate symbols in $PR(l, n, \Pi_0)$ and terms whose lengths are not greater than l, rearrange variables in each axiom, make canonical form w.r.t. Π_0 of them and check whether each obtained EFS is in $\mathcal{MLB}^{[n]}[\Pi_0](l)$ or not by using Lemma 20. □

Theorem 27. *For any $n \geq 0$, the class $\mathcal{LB}^{[\leq n]}$ is theoretical-term-freely and refutably inferable from complete data.*

Proof. Let us consider the procedure in Figure 7, where the procedure read_store is the same one as in Figure 3.

Procedure RIIM $M(n, \Pi_0)$;
begin
  $T = \phi$; $F = \phi$;
  read_store(T, F);
  while $T = \phi$ **do begin** ..... (1)
    output the empty EFS;
    read_store(T, F);
  end;
  $T_0 = T$; $F_0 = F$;
  for $m = 1$ to n **do begin**
    $l_m = \max\{\|A\| \mid A \in T_{m-1}\}$;
    recursively generate $\mathcal{MLB}^{[m]}[\Pi_0](l_m)$, and set it to S;
    for each $\Gamma \in S$ **do** ..... (2)
      while (T, F) is consistent with $M(\Gamma)|_{\Pi_0}$ **do begin** ..... (3)
        output Γ;
        read_store(T, F);
      end;
    $T_m = T$; $F_m = F$;
  end;
  refute the class $\mathcal{LB}^{[\leq n]}$ and stop;
end.

Figure 7: An RIIM for the class $\mathcal{LB}^{[\leq n]}$

Let Π_0 be a nonempty finite subset of Π, and let $I_{\text{base}} \subseteq HB$ be a set of ground atoms. Then assume that we feed a predicate-restricted complete presentation σ of I_{base} w.r.t. Π_0 to the procedure.

(A) In case there is an EFS $\Gamma_{\text{base}} \in \mathcal{LB}^{[\leq n]}$ such that $I_{\text{base}}|_{\Pi_0} = M(\Gamma_{\text{base}})|_{\Pi_0}$.

Claim: In the procedure, for any m with $0 \leq m \leq n$, if T_m and F_m are defined, then (T_m, F_m) is *not* consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[m]}[\Pi_0]$.

Proof of the claim. This proof is given by mathematical induction on m.

(I) In case $m = 0$. It is clear because T_0 is nonempty and the least Herbrand model of the empty EFS is empty.

(II) In case $m \geq 1$. We assume the claim for $m-1$, and assume that T_m and F_m are defined. Then we see that T_{m-1} and F_{m-1} are also defined, and by the induction hypothesis, (T_{m-1}, F_{m-1}) is *not* consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[m-1]}[\Pi_0]$. Therefore by Lemma 25, (T_{m-1}, F_{m-1}) is *not* consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[m]}[\Pi_0] \setminus \mathcal{MLB}^{[m]}[\Pi_0](l_m)$. Thus (T_m, F_m) is *not* consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[m]}[\Pi_0] \setminus \mathcal{MLB}^{[m]}[\Pi_0](l_m)$, because $T_{m-1} \subseteq T_m$ and $F_{m-1} \subseteq F_m$ hold.

Furthermore, since the for-loop (2) terminates, we see that (T_m, F_m) is also *not* consistent with $M(\Gamma)|_{\Pi_0}$ for any EFS $\Gamma \in \mathcal{MLB}^{[m]}[\Pi_0](l_m)$.

Hence we have the claim for m. ■

By Lemma 24, there is an EFS $\Gamma \in \mathcal{MLB}^{[\leq n]}[\Pi_0]$ such that $M(\Gamma)|_{\Pi_0} = M(\Gamma_{\text{base}})|_{\Pi_0}$ $(= I_{\text{base}}|_{\Pi_0})$. Therefore we see by the above claim that T_n and F_n are never defined. By Lemma 14, this means that the procedure outputs an EFS Γ with $M(\Gamma)|_{\Pi_0} = I_{\text{base}}|_{\Pi_0}$ and never terminates the while-loop (1) or (3).

(B) In case there is no EFS $\Gamma \in \mathcal{LB}^{[\leq n]}$ such that $I_{\text{base}}|_{\Pi_0} = M(\Gamma)|_{\Pi_0}$. By using Lemma 14 n times, we see that the procedure refutes the class $\mathcal{LB}^{[\leq n]}$ from σ. □

By Corollary 6, we see that if the number of axioms is not bounded by a constant number, then this class is not refutably inferable, because it contains all nonempty finite concepts on $HB|_{\Pi_0}$.

The following Corollary 28 is obvious from Theorem 27.

Corollary 28. *For any $n \geq 0$, the class of languages definable by EFS's in $\mathcal{LB}^{[\leq n]}$ is refutably inferable from complete data.*

We note that Shinohara[31] showed that the classes in Theorem 27 or Corollary 28 are inferable from positive data.

7. Concluding Remarks

We have pointed out that the essence of the computational logic of scientific discovery or the logic of machine discovery should be the refutability of the whole space of hypotheses by observed data or given facts. Then we have shown a series of such sufficiently large hypothesis spaces in terms of the elementary formal systems. More exactly, for any n the class of length-bounded EFS's with at most n axioms are refutably identifiable. The argument to prove this is also valid for the classes of weakly reducing EFS's, and linear or weakly reducing logic programs[28, 10] with at most n axioms.

The refutability we proposed here forms an interesting contrast to the original one: In the logic of scientific discovery for scientists *each theory* in a hypothesis space is to be *refutable*, while in the computational version of the logic for machines or the logic of machine discovery the *space* itself is to be *refutable* by an observation. Popper contributed to the modern theory of inductive inference. In his books[23, 24], however, Popper strongly denied the induction so far developed by, for example, J. Stuart Mill[19], which had two stages of mechanical creation of a hypothesis from observations and proof of its validity. In fact, he said that the induction, i.e. inference based on many observations, was a myth, and it was neither a psychological fact, nor a fact of ordinary life, nor one of scientific procedure[23].

What he wanted to assert in those books was that scientific theory should have been refuted by observed data or facts and any such theory could by no means be verified. He could not agree with the assertion that the induction should have proved the validity of theory. But we think there were no reasons to deny the stage of mechanical creation of hypotheses.

The inductive inference machine that can refute the hypothesis space itself works as an automatic system for scientific discovery. If the machine for scientific discovery can not refute the whole space of hypotheses, it can just work for *computer aided* scientific discovery. That is, we need to check from time to time whether the machine is still searching for a possible hypothesis.

As a future work we have left the refutability of hypothesis spaces in another two major frameworks of PAC and MAT learning[36, 4], which we will discuss elsewhere. As for the PAC learning we obtained very practical results in the area of Molecular Biology based on our theoretical studies on learnability[8, 9, 29]. Our learning system succeeded in discovering some simple and accurate knowledge, i.e. hypotheses, in a very short time. However it was still an application of PAC *learning*, but *not machine discovery*. In such a system we could not have any way to stop the learning algorithm even when there remains no possibility to find any good hypotheses to explain the data or facts given so far. We were lucky, because the hypotheses our learning system found out happened to be in the spaces we initially gave. The spaces were natural subclasses of the refutably inferable classes we have shown in the last section.

Acknowledgement

The authors wish to thank Efim Kinber for his advice on Theorem 8.

References

[1] D. Angluin, Finding patterns common to a set of strings, in: *Proceedings of the 11th Annual Symposium on Theory of Computing* (1979) 130–141.

[2] D. Angluin, Inductive inference of formal languages from positive data, *Information and Control* **45** (1980) 117–135.

[3] D. Angluin and C.H. Smith, Inductive inference: theory and methods, *ACM Computing Surveys* **15** No. 3 (1983) 237–269.

[4] D. Angluin, Queries and concept learning, *Machine Learning* **2** (1987) 319–342.

[5] S. Arikawa, Elementary formal systems and formal languages – simple formal systems, *Memoirs of Faculty of Science, Kyushu University, Series A, Math.* **24** (1970) 47–75.

[6] S. Arikawa, T. Shinohara and A. Yamamoto, Elementary formal systems as a unifying framework for language learning, in: *Proceedings of the 2nd Annual ACM Workshop on Computational Learning Theory* (1989) 312–327.

[7] S. Arikawa, T. Shinohara and A. Yamamoto, Learning elementary formal systems, *Theoretical Computer Science* **95** (1992) 97–113.

[8] S. Arikawa, S. Kuhara, S. Miyano, A. Shinohara and T. Shinohara, A learning algorithm for elementary formal systems and its experiments on identification of transmembrane domains, in: *Proceedings of the 25th Hawaii International Conference on System Sciences* (1992) 675–684.

[9] S. Arikawa, S. Kuhara, S. Miyano, Y. Mukouchi, A. Shinohara and T. Shinohara, A machine discovery from amino acid sequences by decision trees over regular patterns, *New Generation Computing* **11** No. 3, 4 (1993) 361–375.

[10] H. Arimura, Completeness of depth-bounded resolution for weakly reducing programs, in: *Proceedings of the Software Science and Engineering, World Scientific Series in Computer Science* **31** (1991) 227–245.

[11] L. Blum and M. Blum, Toward a mathematical theory of inductive inference, *Information and Control* **28** (1975) 125–155.

[12] E.M. Gold, Language identification in the limit, *Information and Control* **10** (1967) 447–474.

[13] S. Kapur, Monotonic language learning, in: *Proceedings of the 3rd Workshop on Algorithmic Learning Theory (1992), Lecture Notes in Artificial Intelligence* **743** (1993) 147–158.

[14] E. Kinber, Personal communication, 1993.

[15] S. Lange and T. Zeugmann, Monotonic versus non-monotonic language learning, in: *Proceedings of the Second International Workshop on Nonmonotonic and Inductive Logic (1991), Lecture Notes in Artificial Intelligence* **659** (1993) 254–269.

[16] S. Lange and T. Zeugmann, Types of monotonic language learning and their characterization, in: *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory* (1992) 377–390.

[17] S. Lange and T. Zeugmann, Learning recursive languages with bounded mind changes, *GOSLER-Report 16/92, FB Mathematik und Informatik, TH Leipzig,* September 1992.

[18] J.W. Lloyd, *Foundations of logic programming*, Springer-Verlag, 1984.

[19] J.S. Mill, *Being a connected view of the principles of evidence and the methods of scientific investigation*, Longmans, Green, and Co. 1919.

[20] E. Minicozzi, Some natural properties of strong-identification in inductive inference, *Theoretical Computer Science* **2** (1976) 345–360.

[21] Y. Mukouchi, Characterization of finite identification, in: *Proceedings of the Third International Workshop on Analogical and Inductive Inference, Lecture Notes in Artificial Intelligence* **642** (1992) 260–267.

[22] Y. Mukouchi and S. Arikawa, Inductive inference machines that can refute hypothesis spaces, in: *Proceedings of the Fourth International Workshop on Algorithmic Learning Theory, Lecture Notes in Artificial Intelligence* **744** (1993) 123–136.

[23] R.K. Popper, *Conjectures and refutations*, Routledge and Kegan Paul, 1963.

[24] R.K. Popper, *The logic of scientific discovery*, Harper&Row, 1965.

[25] H. Rogers Jr., *Theory of recursive functions and effective computability*, McGraw-Hill, 1967.

[26] A. Sakurai, Inductive inference of formal languages from positive data enumerated primitive-recursively, in: *Proceedings of the 2nd Workshop on Algorithmic Learning Theory* (1991) 73–83.

[27] E.Y. Shapiro, Inductive inference of theories from facts, *Technical Report 192, Yale University Computer Science Dept.,* 1981.

[28] E.Y. Shapiro, Alternation and the computational complexity of logic programs, *the Journal of Logic Programming* **1** No. 1 (1984) 19–33.

[29] S. Shimozono, A. Shinohara, T. Shinohara, S. Miyano S., S. Kuhara and S. Arikawa, Finding alphabet indexing for decision trees over regular patterns: an approach to bioinformatical knowledge acquisition, in: *Proceedings of the 26th Hawaii International Conference on System Sciences* (1993) 763–772.

[30] T. Shinohara, Inferring unions of two pattern languages, *Bulletin of Informatics and Cybernetics* **20** (1983) 83–87.

[31] T. Shinohara, Inductive inference from positive data is powerful, in: *Proceedings of the Third Annual ACM Workshop on Computational Learning Theory* (1990) 97–110, (*to appear as* Rich classes inferable from positive data: Length-bounded elementary formal systems, *in Information and Computation*).

[32] T. Shinohara, Inductive inference of monotonic formal systems from positive data, in: *Proceedings of the 1st International Workshop on Algorithmic Learning Theory* (1990) 339–351.

[33] J. Shrager and P. Langley (Editors), *Computational models of scientific discovery and theory formation*, Morgan Kaufmann, 1990.

[34] R.M. Smullyan, *Theory of formal systems*, Princeton University Press, 1961.

[35] K. Takashima, *Popperian philosophy*, University of Tokyo Press, 1974 (in Japanese).

[36] L.G. Valiant, A theory of the learnable, *Communications of the ACM* **27** No. 11 (1984) 1134–1142.

[37] K. Wright, Identification of unions of languages drawn from an identifiable class, in: *Proceedings of the 2nd Annual ACM Workshop on Computational Learning Theory* (1989) 328–333.

[38] A. Yamamoto, Procedural semantics and negative information of elementary formal system, *the Journal of Logic Programming* **13** No. 4 (1992) 89–98.