Inductive Inference with Bounded Mind Changes

Mukouchi, Yasuhito
Research Institute of Fundamental Information Science Kyushu University

http://hdl.handle.net/2324/3161

出版情報：RIFIS Technical Report. 58, 1992-05-14. Research Institute of Fundamental Information Science, Kyushu University
バージョン：

RIFIS Technical Report

RIFIS-TR-CS-58

Inductive Inference with Bounded Mind Changes

Yasuhito Mukouchi

May 14, 1992
Revised: September 10, 1992

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. 2329

INDUCTIVE INFERENCE WITH BOUNDED MIND CHANGES

YASUHITO MUKOUCHI†

Department of Information Systems,
Kyushu University 39, Kasuga 816, Japan

Abstract. In this paper, we deal with inductive inference for a class of recursive languages with a bounded number of mind changes. We introduce an n-bounded finite tell-tale and a pair of n-bounded finite tell-tales of a language, and present a necessary and sufficient condition for a class to be inferable with bounded mind changes, when the equivalence of any two languages in the class is effectively decidable. We also show that the inferability of a class from positive data strictly increases, when the allowed number of mind changes increases. In his previous paper, Mukouchi gave necessary and sufficient conditions for a class of recursive languages to be finitely identifiable, that is, to be inferable without any mind changes from positive or complete data. The results we present in this paper are natural extensions of the above results.

1. Introduction

Inductive inference is a process of hypothesizing a general rule from examples. As a correct inference criterion for inductive inference of formal languages and models of logic programming, we have mainly used Gold's identification in the limit[5]. In this criterion, an inference machine is allowed to change its guesses finitely many times, and the guesses are required to converge to a correct guess. Angluin[1], Wright[11] and Sato&Umayahara[6] discussed conditions for a class of formal languages to be inferable from positive data. Shinohara[7, 8] also discussed inductive inferability from positive data in more general setting and exhibited that inductive inference from positive data is much more powerful than it has been believed.

Considering ordinary learning process of human beings, the criterion of identification in the limit seems to be natural. However, we can not decide in general whether a sequence of guesses from an inference machine converges or not at a certain time, and the results of the inference necessarily involve some risks. In his previous paper[10], Mukouchi gave necessary and sufficient conditions for a class of recursive languages to be finitely identifiable, that is, to be inferable without any mind changes from positive or complete data. We use the phrase 'mind change' to mean that an inference machine changes its guess.

†Mailing address: Research Institute of Fundamental Information Science, Kyushu University 33, Fukuoka 812, Japan.

E-mail: mukouchi@rifis.sci.kyushu-u.ac.jp

In this paper, we deal with inductive inference for a class of recursive languages with a bounded number of mind changes. The results we present in this paper are natural extensions of the above results concerning finite identification.

Note that Case&Smith[3] discussed inductive inference of a class of recursive functions from view point of anomalies and mind changes, and showed that there is a natural hierarchy. Case&Lynes[4] also showed that an anomaly hierarchy exists even in case of a class of recursive languages.

In Section 2, we prepare some necessary concepts for our discussions. We also recall the results on finite identification from positive and complete data. In Section 3, we discuss conditions for a class to be inferable with bounded mind changes from positive data. Angluin[1] introduced the notion of a finite tell-tale of a language to discuss inferability of formal languages from positive data, and showed that a class is inferable from positive data if and only if there is a recursive procedure to enumerate all elements in the finite tell-tale of any language of the class. In this paper, we introduce an n-bounded finite tell-tale of a language, and present a necessary and sufficient condition for a class to be inferable with bounded mind changes, when the equivalence of any two languages in the class is effectively decidable. We also exhibit a concrete class of recursive languages which is inferable with at most n mind changes but not inferable with at most $n-1$ mind changes, and show that the inferability of a class strictly increases, when the allowed number of mind changes increases. Case&Smith[3] showed similar results for a class of recursive functions. In Section 4, we give a necessary and sufficient condition for a class to be inferable with bounded mind changes *from complete data*, which is analogous to the above condition concerning positive data.

2. Preliminaries

Let U be a recursively enumerable set to which we refer as a *universal set.* Then we call $L \subseteq U$ a *language.* We do not consider the empty language in this paper.

Definition 2.1. *A class of languages $\Gamma = L_1, L_2, \cdots$ is said to be an indexed family of recursive languages if there exists a computable function $f : N \times U \to \{0,1\}$ such that*

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

From now on, we assume a class of languages is an indexed family of recursive languages without any notice.

Definition 2.2. *A positive presentation of a language L is an infinite sequence $w_1, w_2, \cdots$ of elements of U such that $\{w_1, w_2, \cdots\} = L$.*

A complete presentation of a language L is an infinite sequence $(w_1, t_1), (w_2, t_2), \cdots$ of elements of $U \times \{0,1\}$ such that $\{w_i \mid t_i = 1, i \geq 1\} = L$ and $\{w_j \mid t_j = 0, j \geq 1\} = U - L$.

Positive or complete presentations are denoted by σ, δ, the finite sequence which consists of first $n \geq 0$ data in σ by $\sigma[n]$ and the finite set by $\sigma(n)$.

For a finite sequence $\sigma[n]$ and a sequence δ, the sequence which is obtained by concatenating $\sigma[n]$ with δ is denoted by $\sigma[n] \cdot \delta$.

Definition 2.3. *An n-bounded inference machine (abbreviated to IM_n; $n \geq 0$ or $n = *$) is an effective procedure that requests inputs from time to time and produces outputs from time to time, where if $n \geq 0$, it produces at most $n+1$ outputs, and if $n = *$, it produces at most finitely many outputs.*

The outputs produced by the machine are called guesses.

For a finite sequence $\sigma[m] = w_1, w_2, \ldots, w_m$, we denote by $M(\sigma[m])$ the last guess produced by an IM_n M which is successively fed $w_1, w_2, \ldots, w_m$ on its input requests.

The inference machines we are dealing with in this paper may not produce a guess after reading a datum until requesting a next datum.

Definition 2.4. *A class $\Gamma = L_1, L_2, \cdots$ is said to be EX_n identifiable from positive data (resp., complete data) if there exists an IM_n M satisfying the following ($n \geq 0$ or $n = *$): For any language L_i of Γ and for any positive presentation (resp., complete presentation) σ of L_i, the last guess k of M which is successively fed σ's data satisfies $L_k = L_i$.*

A class Γ is said to be finitely identifiable (resp., identifiable in the limit) if it is EX_0 identifiable (resp., EX_* identifiable).

A class Γ is also said to be EX-TXT_n identifiable (resp., EX-INF_n identifiable) if it is EX_n identifiable from positive data (resp., complete data). By the same notation EX-TXT_n (resp., EX-INF_n), we also denote the set of the classes that are EX-TXT_n identifiable (resp., EX-INF_n identifiable).

In this paper, a finite-set-valued function F is said to be *computable* if there exists an effective procedure that produces all elements in $F(x)$ and then halts uniformly for any argument x.

Mukouchi[10] presented necessary and sufficient conditions for an indexed family of recursive languages to be finitely identifiable.

Definition 2.5 (Mukouchi[10]). *A set S_i is said to be a definite finite tell-tale of L_i if*

(1) S_i is a finite subset of L_i, and
(2) $S_i \subseteq L_j$ implies $L_j = L_i$ for any index j.

Theorem 2.1 (Mukouchi[10]). *A class Γ is finitely identifiable from positive data if and only if a definite finite tell-tale of L_i is uniformly computable for any index i, that is, there exists an effective procedure that on input i produces all elements of a definite finite tell-tale of L_i and then halts.*

Definition 2.6 (Mukouchi[10]). *A language L is said to be consistent with a pair of sets $\langle T, F\rangle$ if $T \subseteq L$ and $F \subseteq U - L$.*

A pair of sets $\langle T_i, F_i\rangle$ is said to be a pair of definite finite tell-tales of L_i if

(1) T_i is a finite subset of L_i,
(2) F_i is a finite subset of $U - L_i$, and
(3) if L_j is consistent with the pair $\langle T_i, F_i\rangle$, then $L_j = L_i$.

Theorem 2.2 (Mukouchi[10]). *A class Γ is finitely identifiable from complete data if and only if a pair of definite finite tell-tales of L_i is uniformly computable for any index i.*

The following corollary shows a necessary condition for a class to be finitely identifiable.

Corollary 2.3. *If a class Γ is finitely identifiable from positive or complete data, then whether $L_i = L_j$ or not is effectively decidable for any indices i, j.*

Proof: Clearly, if Γ is finitely identifiable from positive data, then Γ is also finitely identifiable from complete data. Therefore, it suffices to show the case of complete data.

Suppose Γ is finitely identifiable from complete data. Fix arbitrary indices i, j. To begin with, compute a pair of definite finite tell-tales of L_i, and set it to $\langle T_i, F_i \rangle$. We can effectively compute this pair by Theorem 2.2. Then check whether L_j is consistent with $\langle T_i, F_i \rangle$. We can effectively check this, because T_i and F_i are explicitly given finite sets. If L_j is not consistent with $\langle T_i, F_i \rangle$, then we conclude $L_i \neq L_j$, because L_i is consistent with $\langle T_i, F_i \rangle$. Otherwise, we conclude $L_i = L_j$ by Definition 2.6. ∎

3. Inductive Inference with Bounded Mind Changes from Positive Data

First of all, we give a necessary condition for a class Γ to be $EX\text{-}TXT_n$ identifiable.

Proposition 3.1. *For any $n \geq 1$, if a class Γ contains languages $L_{i_0}, L_{i_1}, \ldots, L_{i_n}$ such that $L_{i_0} \subsetneq L_{i_1} \subsetneq \cdots \subsetneq L_{i_n}$, then Γ is not $EX\text{-}TXT_{n-1}$ identifiable.*

Proof: Suppose that Γ contains languages $L_{i_0}, L_{i_1}, \ldots, L_{i_n}$ such that $L_{i_0} \subsetneq L_{i_1} \subsetneq \cdots \subsetneq L_{i_n}$ and that Γ is $EX\text{-}TXT_*$ identifiable by an IM_* M. For simplicity, put $L'_j = L_{i_j}$ $(0 \leq j \leq n)$. We show that M needs to change its guesses more than n times to identify L'_n from a certain positive presentation of L'_n. Let σ_j be an arbitrary positive presentation of L'_j. We recursively define c_j and δ_j as follows:

Stage 0:

Let $c_0 := 0$ and $\delta_0 := \sigma_0$. Goto Stage 1.

Stage $m (1 \leq m \leq n)$:

Let

$$c_m := \min\{c > c_{m-1} \mid \exists k \text{ s.t. } M(\delta_{m-1}[c]) = k \wedge L'_{m-1} = L_k\} \quad \text{and}$$
$$\delta_m := \delta_{m-1}[c_m] \cdot \sigma_m.$$

Such an integer c_m exists, because δ_{m-1} is a positive presentation of L'_{m-1} and Γ is $EX\text{-}TXT_*$ identifiable by M. Note that the above δ_m becomes a positive presentation of L'_m, because $L'_{m-1} \subsetneq L'_m$.

Goto Stage $m + 1$.

Stage $n + 1$:

Let

$$c_{n+1} := \min\{c > c_n \mid \exists k \text{ s.t. } M(\delta_n[c]) = k \wedge L'_n = L_k\}.$$

When we feed a positive presentation δ_n of L'_n successively to M, it should output guesses after reading c_1-th, c_2-th, ..., c_{n+1}-th datum, and so it can not identify L'_n within n mind changes. ∎

Before proceeding to the next corollary, we briefly recall a pattern and a pattern language. (For more details, see Angluin[2] or Mukouchi[9].)

Fix a finite alphabet with at least two constant symbols. 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 of 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 effectively decidable. Therefore, we can consider the class of pattern languages as an indexed family of recursive languages, where the pattern itself is considered to be an index.

Corollary 3.2. *For any $n \geq 0$, the class of pattern languages is not EX-TXT_n identifiable.*

Proof: By Proposition 3.1, it suffices to show that there exist patterns $\pi_0, \pi_1, \ldots, \pi_m$ such that $L(\pi_0) \subsetneq L(\pi_1) \subsetneq \cdots \subsetneq L(\pi_m)$ for any $m \geq 1$.

In fact, let $\pi_0 = x_1 x_2 \cdots x_{m+1}, \pi_1 = x_1 x_2 \cdots x_m, \ldots, \pi_m = x_1$. Then $L(\pi_i)$ is the set of all constant strings of length more than $m - i$, and clearly

$$L(\pi_0) \subsetneq L(\pi_1) \subsetneq \cdots \subsetneq L(\pi_m). \quad \blacksquare$$

Angluin[1] showed that the class of pattern languages is inferable from positive data in the limit, that is, it is EX-TXT_* identifiable.

Definition 3.1. *Let $\Gamma = L_1, L_2, \cdots$. A set S_i is said to be a 0-bounded finite tell-tale (abbreviated to FT_0) in Γ of L_i if S_i is a definite finite tell-tale of L_i, that is,*

(1) S_i is a finite subset of L_i, and

(2) $S_i \subseteq L_j$ implies $L_j = L_i$ for any index j.

A set S_i is said to be an n-bounded finite tell-tale (abbreviated to FT_n ; $n \geq 1$) in Γ of L_i if

(1) S_i is a finite subset of L_i, and

(2) if $L_j \neq L_i$ and $S_i \subseteq L_j$, then there exist an FT_{n-1} in Γ of L_j.

Intuitively, an FT_n in Γ of L_i is a tell-tale which validates producing the guess i, when the inference machine is allowed to produce another $n - 1$ guesses.

We can easily prove by induction on n that if a certain finite set S is an FT_n in Γ of L_i, then S is also an FT_{n+1} in Γ of L_i ($n \geq 0$).

Definition 3.2. *An FT_0 in Γ of L_i is said to be recurrently computable if a certain FT_0 S_i in Γ of L_i is computable.*

An FT_n in Γ of L_i is said to be recurrently computable ($n \geq 1$) if

(1) a certain FT_n S_i in Γ of L_i is computable, and

(2) for any index j, if $L_j \neq L_i$ and $S_i \subseteq L_j$, then an FT_{n-1} in Γ of L_j is recurrently computable.

An FT_n of Γ is said to be recurrently constructible if an FT_n in Γ of L_i is recurrently computable for any index i ($n \geq 0$).

Lemma 3.3. *Suppose whether $L_i = L_j$ or not is effectively decidable for any indices i, j.*

For any $n \geq 0$, if a class Γ is EX-TXT$_n$ identifiable, then an FT_n of Γ is recurrently constructible.

Proof: Suppose Γ is *EX-TXT$_n$* identifiable by an *IM$_n$* M. In what follows, for a finite sequence φ, we denote the corresponding finite set by $\tilde{\varphi}$.

We consider the following partial recursive procedure which produces a finite sequence of U:

Procedure $Ft(m, i)$;
begin
  for each finite sequence ψ of U **do**
    /* Note that all finite sequences of U are recursively enumerable */
   if $\tilde{\psi} \subseteq L_i$ **do begin**
    Initialize M;
    Feed successively ψ to M on its input requests;
    if M produces any guess **then begin**
     Let k be the number of guesses ($1 \leq k \leq n+1$) and
       g be the last guess produced by M;
     if $(k > n - m) \wedge (L_g = L_i)$ **then** output ψ and stop;
    end;
   end;
end.

(1) Clearly, if $Ft(m, i)$ is defined, then $\tilde{Ft}(m, i)$ is a finite subset of L_i.

(2) If $Ft(0, i)$ is defined, then $\tilde{Ft}(0, i)$ is an FT_0 of L_i. In fact, suppose that $Ft(0, i)$ is defined and that $\tilde{Ft}(0, i)$ is not an FT_0 of L_i. Then, there exists an index j such that $L_j \neq L_i$ and $\tilde{Ft}(0, i) \subseteq L_j$. Let σ_j be an arbitrary positive presentation of L_j. Since $Ft(0, i) \cdot \sigma_j$ is a positive presentation of L_j, it follows that M can not identify L_j from $Ft(0, i) \cdot \sigma_j$ with at most n mind changes, which contradicts the assumption.

(3) For any m ($0 < m \leq n$) and any index i, if $Ft(m, i)$ is defined and there exists an index j such that $L_j \neq L_i$ and $\tilde{Ft}(m, i) \subseteq L_j$, then $Ft(m-1, j)$ is defined. In fact, suppose $Ft(m-1, j)$ is not defined. Let σ_j be an arbitrary positive presentation of L_j and $\delta = Ft(m, i) \cdot \sigma_j$. Then there exists a $k > |Ft(m, i)|$ such that $M(\delta[k]) = g$ and $L_g = L_j$ for some g. Since $\delta[k]$ is a finite sequence of U, it should appear in the for loop above. Furthermore, when M is successively fed $\delta[k]$ on its input requests, it should produce more than $n - m + 1$ guesses. Hence the procedure will produce an output. This is a contradiction.

By (1), (2) and (3), for any m ($0 < m \leq n$) and any index i, if $Ft(m, i)$ is defined, then it is an FT_m in Γ of L_i.

Moreover, for any index i, $Ft(n, i)$ is defined, since Γ is identifiable by M. Therefore, an FT_n of Γ is recurrently constructible. ∎

Lemma 3.4. *Suppose whether $L_i = L_j$ or not is effectively decidable for any indices i, j.*

For any $n \geq 0$, if an FT_n of a class Γ is recurrently constructible, then Γ is EX-TXT_n identifiable.

Proof: Suppose an FT_n of Γ is recurrently constructible. We denote by $FT_m(i)$ the result of computation of an FT_m in Γ of L_i ($m \geq 0$). We consider the following procedure:

```
Procedure M;
begin
    m := n;    k := 0;
    S := φ;    T := φ;
    for j := 1 to ∞ do begin
        read a next datum and add it to T;        /* Note that ♯T = j */
        for i := 1 to j do
            if (k = 0) ∨ (L_k ≠ L_i ∧ S ⊆ L_i) then
                if FT_m(i) ⊆ T then begin
                    output i;
                    if m = 0 then stop;
                    S := FT_m(i);    k := i;
                    m := m − 1;
                end;
    end;
end.
```

Clearly, this procedure produces at most $n+1$ outputs. Suppose we are going to feed a positive presentation σ successively to the procedure on its input requests.

(1) This procedure produces at least one guess. In fact, suppose this procedure never produces a guess. When it reaches the case

$$j = \max\{h, \min\{l \mid FT_n(h) \subseteq \sigma(l)\}\} \quad \text{and} \quad i = h,$$

this procedure should produce the guess h, which contradicts the assumption.

(2) Suppose the last guess, say g, produced by this procedure is not correct.

(i) In case of $m = 0$, when the procedure produced the last guess. It contradicts the definition of an FT_0.

(ii) Otherwise, note that $S \subseteq T$ and $L_g \neq L_h$. When it reaches the case

$$j \geq \max\{h, \min\{l \mid FT_n(h) \subseteq \sigma(l)\}\} \quad \text{and} \quad i = h,$$

this procedure should produce the next guess h, which contradicts the assumption. ∎

We obtain the following Theorem 3.5 by Lemma 3.3 and Lemma 3.4.

Theorem 3.5. *Suppose whether $L_i = L_j$ or not is effectively decidable for any indices i, j.*

For any $n \geq 0$, a class Γ is EX-TXT_n identifiable if and only if an FT_n of Γ is recurrently constructible.

Note that in case of $n = 0$, the above theorem is equivalent to Theorem 2.1 by Corollary 2.3.

Example 3.1. *Fix an arbitrary number $n \geq 0$. We consider the following set:*

$$C_n = \left\{ (q_1, q_2, \ldots, q_{n+1}) \in N^{n+1} \;\middle|\; \begin{array}{c} q_1, q_2, \ldots, q_{n+1} \text{ are prime numbers with} \\ q_1 < q_2 < \cdots < q_{n+1} \end{array} \right\}.$$

Fix an arbitrary computable bijection from C_n to N, and we denote it by $\langle\langle\ \rangle\rangle$. We consider the following class:

$$\Gamma = L_1, L_2, \cdots,$$

where

$$L_{\langle\langle q_1, q_2, \ldots, q_{n+1} \rangle\rangle} = \{m \in N \mid m \text{ is a multiple of } q_j \text{ for some } j\ (1 \leq j \leq n+1)\}.$$

Clearly, this class Γ is an indexed family of recursive languages. This class is also finitely identifiable. In fact, there exists a computable FT_0 of $L_{\langle\langle q_1, q_2, \ldots, q_{n+1} \rangle\rangle}$ such as

$$\{q_1, q_2, \ldots, q_{n+1}\}.$$

Note that if $n \geq 1$, this class Γ does not have the property of so-called finite thickness[1, 11], which is a sufficient condition to be inferable from positive data in the limit.

Example 3.2. *Fix an arbitrary number $n \geq 0$. We consider the following set:*

$$D_n = \left\{ (q_1, q_2, \ldots, q_k) \in N^k \;\middle|\; \begin{array}{c} 1 \leq k \leq n+1, \\ q_1, q_2, \ldots, q_k \text{ are prime numbers with} \\ q_1 < q_2 < \cdots < q_k \end{array} \right\}.$$

Fix an arbitrary computable bijection from D_n to N, and we denote it by $[\![\ ]\!]$. We consider the following class:

$$\Gamma' = L_1, L_2, \cdots,$$

where

$$L_{[\![q_1, q_2, \ldots, q_k]\!]} = \{m \in N \mid m \text{ is a multiple of } q_j \text{ for some } j\ (1 \leq j \leq k)\} \quad (1 \leq k \leq n+1).$$

Clearly, this class Γ' is an indexed family of recursive languages. This class is also EX-TXT$_n$ identifiable. In fact, there exists a computable FT_m $(n - k + 1 \leq m \leq n)$ of $L_{[\![q_1, q_2, \ldots, q_k]\!]}$ such as

$$FT_{n-k+1} = \cdots = FT_n = \{q_1, q_2, \ldots, q_k\},$$

and it follows that an FT_n of Γ' is recurrently constructible.

On the other hand, this class is shown to be not EX-TXT$_{n-1}$ identifiable by Proposition 3.1 if $n \geq 1$.

Note that this class Γ' does not have the property of finite thickness if $n \geq 1$.

From Corollary 3.2, Example 3.2 and the fact that the class of pattern languages is *EX-TXT*$_*$ identifiable but not *EX-TXT*$_n$ identifiable for any $n \geq 0$, we see that there exists a hierarchy such as

$$EX\text{-}TXT_0 \subsetneq EX\text{-}TXT_1 \subsetneq \cdots \subsetneq EX\text{-}TXT_n \subsetneq \cdots \subsetneq EX\text{-}TXT_*.$$

4. Inductive Inference with Bounded Mind Changes from Complete Data

In this section, we give the results concerning complete data, which are analogous to the results in Section 3 concerning positive data.

The following Definition 4.1 and Theorem 4.1 form a remarkable contrast to Definition 3.1 and Theorem 3.5 concerning positive data.

Definition 4.1. *Let $\Gamma = L_1, L_2, \cdots$. A pair $\langle T_i, F_i \rangle$ is said to be a* pair of 0-bounded finite tell-tales *(abbreviated to PFT_0) in Γ of L_i if $\langle T_i, F_i \rangle$ is a pair of definite finite tell-tales of L_i, that is,*

(1) T_i is a finite subset of L_i,

(2) F_i is a finite subset of $U - L_i$, and

(3) for any index j, if L_j is consistent with the pair $\langle T_i, F_i \rangle$, then $L_j = L_i$.

A pair $\langle T_i, F_i \rangle$ is said to be a pair of n-bounded finite tell-tales *(abbreviated to PFT_n ; $n \geq 1$) in Γ of L_i if*

(1) T_i is a finite subset of L_i,

(2) F_i is a finite subset of $U - L_i$, and

(3) for any index j, if $L_j \neq L_i$ and L_j is consistent with $\langle T_i, F_i \rangle$, then there exists a PFT_{n-1} in Γ of L_j.

We can easily prove by induction on n that if a certain pair of finite sets $\langle T, F \rangle$ is a PFT_n in Γ of L_i, then $\langle T, F \rangle$ is also a PFT_{n+1} in Γ of L_i ($n \geq 0$).

Similarly to Definition 3.2, we define the recurrent computability and the recurrent constructibility of a PFT_n.

We can also prove the following theorem in a similar way to the proofs of Lemma 3.3 and Lemma 3.4.

Theorem 4.1. *Suppose whether $L_i = L_j$ or not is effectively decidable for any indices i, j.*

For any $n \geq 0$, a class Γ is EX-INF$_n$ identifiable if and only if a PFT_n of Γ is recurrently constructible.

Note that in case of $n = 0$, the above theorem is equivalent to Theorem 2.2 by Corollary 2.3.

5. Concluding Remarks

We have investigated some characterization theorems on language learning with a bounded number of mind changes, and exhibited necessities of mind changes.

We have also recognized again the importance of paying attention to a characteristic subset of each language in the class, when we consider language learning.

Acknowledgements

The author wishes to thank Prof. Setsuo Arikawa for many suggestions, valuable comments and productive discussions. He also would like to thank Prof. Masako Sato and Mr. Takashi Moriyama at University of Osaka Prefecture for finding errors in earlier version of this paper.

References

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

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

[3] Case, J. and Lynes, C., *Machine inductive inference and language identification*, Proc. of the International Colloquium on Automata, Languages and Programming(ICALP) (1982), 107–115.

[4] Case, J. and Smith, C., *Comparison of identification criteria for machine inductive inference*, Theoretical Computer Science **25** (1983), 193–220.

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

[6] Sato, M. and Umayahara, K., *Inductive inferability for formal languages from positive data*, Proc. 2nd Workshop on Algorithmic Learning Theory (1991), 84–92.

[7] Shinohara, T., *Inductive inference from positive data is powerful*, Proc. 3rd Workshop on Comput. Learning Theory (1990), 97–110.

[8] Shinohara, T., *Inductive inference of monotonic formal systems from positive data*, Proc. 1st Workshop on Algorithmic Learning Theory (1990), 339–351.

[9] Mukouchi, Y., *Characterization of pattern languages*, Proc. 2nd Workshop on Algorithmic Learning Theory (1991), 93–104.

[10] Mukouchi, Y., *Characterization of finite identification*, to appear in Proc. 3rd International Workshop on Analogical and Inductive Inference (1992).

[11] Wright, K., *Identification of unions of languages drawn from an identifiable class*, Proc. 2nd Workshop on Comput. Learning Theory (1989), 328–333.