A note on iteratively working strategies in inductive inference

宮原, 哲浩
九州大学教養部情報科学教室

https://doi.org/10.15017/1449070

出版情報：九州大学教養部数学雑誌. 17 (2), pp.87-91, 1990-12. 九州大学教養部数学教室

Math. Rep.
17-2, 1990.

A note on iteratively working strategies in inductive inference

Tetsuhiro MIYAHARA

(Received September 21, 1990)

1. Introduction

Inductive inference of recursive functions from imput–output examples is considered. An iteratively working strategy utilizes the last hypothesis produced by it and the present example. We investigate the inferring power of some iteratively working strategies with anomalies.

A process of automatic program synthesis can be formalized to inductive inference of recursive functions from examples as an infinite process. Its situation can be imagined as follows: An algorithmic device, which is formally a partial recursive function and called a *strategy* or an *inductive inference machine*, takes as the input the graph of a recursive function f which is the list of all examples $(x, f(x))$ for natural number x. As it receives the list, it produces infinitely many computer programs called *hypotheses*. When almost all programs produced by a strategy are equal to a program that computes f, we say that the strategy *inductively infers* (or *identifies*) f. A set of recursive functions is said to be identifiable by a strategy if the strategy identifies every function in the set.

There are many possible requirements, called *identification criteria*, on the process of synthesizing programs and the sequence of programs produced by a strategy. The power of an identification criterion, called an *identification type*, is expressed by the class of all sets of recursive functions each of which can be identified by some strategy under the identification criterion.

2. Identification types

We give some basic definitions and notations and present some fundamental results. $\mathfrak{N}=\{0, 1, 2, \ldots\}$ denotes the set of all natural numbers and $\mathfrak{N}^*$ denotes the set of all finite sequences of natural numbers. $\subseteq$ and $\subset$ denote containment and proper containment for sets, respectively. A function will mean a function of one variable unless otherwise indicated. The classes of all partial recursive and (total) recursive functions are denoted by $\mathcal{P}$ and $\mathcal{R}$, respectively. A function from $\mathcal{R}$ is sometimes identified with the sequence of its values. Let $\langle \ldots \rangle$ be a fixed effective encoding of $\mathfrak{N}^*$. $f(x)\downarrow$ means that a partial recursive function f is defined on a natural number x. $f[k]$ denotes $\langle f(0), f(1), \ldots, f(k)\rangle$, where f is a partial recursive function and $f(0), f(1), \ldots, f(k)$ $(k \geq 0)$ are defined. A sequence $(h_k)_{k\in\mathfrak{N}}$ of natural numbers is said to converge to a natural number p, denoted by $lim_k h_k=p$, iff all but finitely many elements in the sequence are identical to p. Let $\varphi_0, \varphi_1, \varphi_2, \ldots$ be a fixed acceptable programming system of $\mathcal{P}$ [4]. p is said to be a program for f, if $\varphi_p=f$.

Let $p\in\mathfrak{N}$ and $f\in\mathcal{R}$. $\varphi_p(x)\neq f(x)$ means that $\varphi_p(x)$ is defined and not equal to $f(x)$, or $\varphi_p(x)$ is undefined. Then x is called an *anomaly*. We suppose n ranges over $\mathfrak{N}$. We wright $\varphi_p=^n f$ if the cardinality of $\{x\in\mathfrak{N} \mid \varphi_p(x)\neq f(x)\}\leq n$. By suitable encoding, we may treat a strategy or an inductive inference machine M as a partial recursive function. At time k an inductive inference machine M takes as input $f[k]$ which is a coding of a finite sequence $f(0), f(1), \ldots, f(k)$ of data. M's hypothesis is a number and $M(f[k])$ denotes the hypothesis at time k, taken the data of f as an input. First we review the Gold's paradigm of *identification in the limit* or *EX-identification*.

DEFINITION 1 [1, 2] $M\in\mathcal{P}$ *EX-identifies* $f\in\mathcal{R}$ iff $\forall k\in\mathfrak{N}$, $M(f[k])\downarrow$ and there exists $p=lim_k M(f[k])$ and $\varphi_p=f$. We define as follows: $EX(M)=\{f\in\mathcal{R} \mid M\in\mathcal{P}$ EX-identifies $f\}$, and $EX=\{U\subseteq\mathcal{R} \mid \exists M\in\mathcal{P};\ U\subseteq EX(M)\}$.

Thus EX represents the power of EX-identification and called an identification type. Other identification types are defined similarly. We consider criteria allowing the last hypothesis to have finitely many anomalies.

DEFINITION 2 [1] $M\in\mathcal{P}$ *EX^n-identifies* $f\in\mathcal{R}$ iff $\forall k\in\mathfrak{N}$, $M(f[k])\downarrow$ and there exists $p=lim_k M(f[k])$ and $\varphi_p =^n f$.

By the definitions 1 and 2, EX^0 coincides with EX and the similar relations hold for other identification types with anomalies.

For an inductive inference machine M to EX^n-identify a function f, we allow M to utilize the all data $f(0)$, $f(1)$, . . . , $f(k)$ at time k. Since an inductive inference is an infinite process, an inductive inference machine under no restriction on the amount of accessible input data is not realistic. Thus we have an iteratively working machine M that iterates a process of making a hypothesis h_{k+1} based on the its last hypothesis h_k and the present example $(k, f(k))$ at time k.

The inferring power of some iteratively working machines are investigated in [3, 5, 6, 7]. We consider an extension of such machine in the next section. In the definitions 3, 7 and 8, the hypothesis $h_k(f)$, if any, is defined as follows: $h_0(f)=0$, and $h_{k+1}(f)=M(h_k(f), k, f(k))$, $(k\geq 0)$, where M is a partial recursive function of three variables and f is a recursive function.

Definition 3 [3, 5, 7] Let M be a partial recursive function of three variables. *M IT^n-identifies* $f\in\mathfrak{R}$ iff $\forall k\in\mathfrak{N}$, $h_k(f)$ is defined, and there exists $p=lim_k h_k(f)$ and $\varphi_p =^n f$.

In the criteria described so far, we require the infinite sequence of the hypotheses output by an inductive inference machine to converges as a sequence of natural numbers. This means that the sequence of programs represented by the hypotheses converges syntactically.

According to the approach in [1], we remove this requirement and consider the FEX^n, OEX^n and BC-identifications.

Definition 4 [1] *$M\in\mathfrak{P}$ FEX^n-identifies* $f\in\mathfrak{R}$ iff $\forall k\in\mathfrak{N}$, $M(f[k])\downarrow$ and $\{M(f[k]) \mid k\in\mathfrak{N}\}$ is a finite set and if p occurs infinitely often in the sequence$(M(f[k]))_{k\in\mathfrak{N}}$ then $\varphi_p =^n f$.

Definition 5 [1] *$M\in\mathfrak{P}$ OEX^n-identifies* $f\in\mathfrak{R}$ iff $\forall k\in\mathfrak{N}$, $M(f[k])\downarrow$ and

$\{M(f[k]) \mid k \in \mathfrak{N}\}$ is a finite set and there exists p that occurs infinitely often in the sequence$(M(f[k]))_{k\in\mathfrak{N}}$ and $\varphi_p =^n f$.

DEFINITION 6 [1] $M \in \mathfrak{P}$ *BC-identifies* $f \in \mathfrak{R}$ iff $\forall k \in \mathfrak{N}$, $M(f[k])\downarrow$ and $\exists k_0 \in \mathfrak{N}$; $\forall k \geq k_0$, $\varphi_{M(f[k])} = f$.

In the following theorem we summarize a part of results, which is concerned with this note.

THEOREM 1 [1, 7]
(1) $IT \subset EX$.
(2) $EX^n = FEX^n = OEX^n$ $(n \in \mathfrak{N})$.
(3) $EX \subset BC$.

3. An extension of iteratively working strategies

The definition of IT^n-identification allows the coding of input data to a hypothesis by a padding function [4]. If we modify the definition 3 as follows: M *IT′-identifies* $f \in \mathfrak{R}$ iff $\forall k \in \mathfrak{N}$, $h_k(f)$ is defined, and there exists $k_0 \in \mathfrak{N}$ such that $\forall k \geq k_0$, $\varphi_{h_k(f)} = f$, then we have $IT' = BC$. Thus we consider some criteria allowing iteratively working machines to code the input data to a hypothesis to some extent. These criteria correspond to FEX^n and OEX^n-identifications.

DEFINITION 7 $M \in \mathfrak{P}$ *FIT^n-identifies* $f \in \mathfrak{R}$ iff $\forall k \in \mathfrak{N}$, $h_k(f)$ is defined and $\{h_k(f) \mid k \in \mathfrak{N}\}$ is a finite set and if p occurs infinitely often in the sequence $(h_k(f))_{k\in\mathfrak{N}}$ then $\varphi_p =^n f$.

DEFINITION 8 $M \in \mathfrak{P}$ *OIT^n-identifies* $f \in \mathfrak{R}$ iff $\forall k \in \mathfrak{N}$, $h_k(f)$ is defined and $\{h_k(f) \mid k \in \mathfrak{N}\}$ is a finite set and there exists p that occurs infinitely often in the sequence $(h_k(f))_{k\in\mathfrak{N}}$ and $\varphi_p =^n f$.

By the above definitions the following relations hold: $EX^n \subseteq FEX^n \subseteq OEX^n$, $IT^n \subseteq FIT^n \subseteq OIT^n$, $IT^n \subseteq EX^n$, $FIT^n \subseteq FEX^n$, and $OIT^n \subseteq OEX^n$. By the Theorem 1 (2), we have $IT^n \subseteq FIT^n \subseteq OIT^n \subseteq EX^n$, and by Theorem 1 (1), at least

Thus we have $h_j(g^1)=h_j(g^2)$ $(0\leq j\leq ml+1,\ (m+1)l\leq j)$. Since $g^1, g^2\in U_0\subseteq U\subseteq IT^n(M)$, there exists a number m such that

$$p=lim_j(g^1)=lim_j(g^2),\ \varphi_p={}^n g^1 \text{ and } \varphi_p={}^n g^2.$$

This contradicts the asumption $l>m$. So there is no number m such that $h^1{}_{(m+1)l+1}=h^2{}_{(m+1)l+1}=h_{\mathrm{m}l+1}(f_i)$, and we have $f_i\in\mathfrak{R}$.

By the recursion theorem, there exists $a>0$ such that $\varphi_a=f_a\in U_1$. Since the sequence $(h_k(f_a))_{k\in\mathfrak{N}}$ of hypotheses output by M taking the data of f_a as input does not converge, we have $f_a\notin IT^n(M)$. This contradicts $f_a\in U_1\subseteq U\subseteq IT^n(M)$. Thus we have $U\notin IT^n$.

References

[1] J. Case, C. Smith: *Comparison of Identification Criteria for Machine Inductive Inference,* Theoretical Computer Science **25** (1983), 193–220.

[2] E. M. Gold: *Language Identification in the Limit,* Information and Control **10** (1967), 447–474.

[3] K. P. Jantke, H.-R. Beick: *Combining Postulates of Naturalness in Inductive Inference,* Elektron. Informationsverarb. Kybernet. **17** (1981), 465–484.

[4] M. Machtey, P. Young: An Introduction to the General Theory of Algorithms, North-Holland, 1978.

[5] T. Miyahara: *Inductive Inference by Iteratively Working and Consistent Strategies with Anomalies,* Bull. Informatics and Cybernetics **22** (1987), 171–177.

[6] D. N. Osherson, M. Stob, S. Weinstein: Systems That Learn-An Introduction to Learning Theory for Cognitive and Computer Scientists, The MIT Press, 1986.

[7] R. Wiehagen: *Limes-Erkennung rekursiver Funktionen durch Spezielle Strategien,* Elektron. Informationsverarb. Kybernet. **12** (1976), 93–99.

one of three containments in the relation $IT \subseteq FIT \subseteq OIT \subseteq EX$ is proper.

We show that an extension of theorem 1 (1) holds by using the same technique in [7]. Therefore at least one of three containments in the relation $IT^n \subseteq FIT^n \subseteq OIT^n \subseteq EX^n$ is proper.

THEOREM 2 $IT^n \subset EX^n$ $(n \in \mathfrak{N})$.

PROOF. We show that there exists $U \in EX^n \backslash IT^n$. Let $U_0 = \{\alpha 0^\infty \mid \alpha \in \mathfrak{N}^*\}$, $U_1 = \{f \in \mathfrak{R} \mid \varphi_{f(0)} = f,\ \forall x \in \mathfrak{N},\ f(x) \neq 0\}$ and $U = U_0 \cup U_1$. Obviously $U \in EX^n$. We assume $U \in IT^n$. There exists a partial recursive function M of three variables such that $U = U_0 \cup U_1 \subseteq IT^n(M)$.

Let $l > 2n$ and $q = 1, 2$ and $i \in \mathfrak{N}$. We define f_i as follows:
Let

$$f_i(0) = i,$$
$$h_0(f_i) = 0,$$
$$h_1(f_i) = M(h_0(f_i),\ 0,\ f_i(0)).$$

For each $m = 0, 1, 2, \ldots,$ we repeat the following process:

$$h^q_{ml+2} = M(h_{ml+1}(f_i),\ ml+1,\ q),$$
$$h^q_{ml+3} = M(h^q_{ml+2},\ ml+2,\ q),$$
$$\vdots$$
$$h^q_{(m+1)l+1} = M(h^q_{(m+1)l},\ (m+1)l,\ q),$$

$$\underset{(ml+1 \leq x \leq (m+1)l)}{f_i(x)} = \begin{cases} 1 & \text{if } h^1_{(m+1)l+1} \neq h_{ml+1}(f_i) \\ 2 & \text{if } h^1_{(m+1)l+1} = h_{ml+1}(f_i) \\ & \quad \text{and } h^2_{(m+1)l+1} \neq h_{ml+1}(f_i), \end{cases}$$

$$h_{ml+2}(f_i) = M(h_{ml+1}(f_i),\ ml+1,\ f_i(ml+1)),$$
$$h_{ml+3}(f_i) = M(h_{ml+2}(f_i),\ ml+2,\ f_i(ml+2)),$$
$$\vdots$$
$$h_{(m+1)l+1}(f_i) = M(h_{(m+1)l}(f_i),\ (m+1)l,\ f_i((m+1)l)).$$

Suppose there is a number m such that $h^1_{(m+1)l+1} = h^2_{(m+1)l+1} = h_{ml+1}(f_i)$.

We define the functions g^1 and g^2 as follows:

$$g^1(x) = \begin{cases} f_i(x) & (0 \leq x \leq ml) \\ 1 & (ml+1 \leq x \leq (m+1)l) \\ 0 & ((m+1)l < x), \end{cases}$$

$$g^2(x) = \begin{cases} f_i(x) & (0 \leq x \leq ml) \\ 2 & (ml+1 \leq x \leq (m+1)l) \\ 0 & ((m+1)l < x). \end{cases}$$