Consistency problem for one-variable patterns is polynomially decidable

Hiroshi Sakamoto

Department of Informatics, Kyushu University, Fukuoka 812-8581, Japan

hiroshi@i.kyushu-u.ac.jp

Abstract

The present paper deals with the decision problem for the class of one-variable patterns, called the consistency problem. Although this problem is obviously in NP, its tractability is unknown. We prove the consistency problem to be decidable in P.

1 Introduction

A pattern is a string from a constant alphabet Σ and a variable alphabet X. The language generated by a pattern π is the set of all constant strings obtained by substituting nonempty strings for the variables of π [1]. A pattern π is said to be a k-variable pattern if at most k different $x_i \in X$ appears in π. The language generated by a pattern π is denoted by $L(\pi)$. A string w is called a positive example of a pattern π if $w \in L(\pi)$. In particular, a pattern π is called descriptive for a finite set S of strings if $S \subseteq L(\pi)$ and for any other pattern π' such that $S \subseteq L(\pi')$, $L(\pi') \not\subseteq L(\pi)$. The problem of finding a descriptive pattern from a given set is referred as to the *pattern inference from positive data*. The problem becomes to one of central issues on Theory of Computing and Machine Learning during the last decade.

Angluin [1] studied computational complexity and several closure properties of patterns. One of the open problems in her paper is known to be the inclusion problem. Recently, the problem became known to be undecidable [8]. After that, a number of researchers have been investigated the pattern inference from positive data with respect to many paradigms.

Marron [13] considered a model in which a single positive example is given and a membership oracle is assumed. He showed that k-variable patterns can be identified using polynomially many membership queries in his model. Arimura et al [3, 15] discussed inferability of unions of pattern languages from positive data. They showed several nontrivial classes of unbounded unions of pattern languages to be inferable from positive data. Lange and Wiehagen [12] studied learning all pattern languages in the limit in polynomial update time. Erlebach et al [4] investigated inferability of one-variable pattern languages in the limit and they designed very efficient algorithm.

In parallel with the pattern inference from positive data, there is another continuous stream. A string w is called a negative example of a π if $w \notin L(\pi)$. When two finite sets of strings are arbitrary given, by assuming that one of the sets is positive data and the other is negative data, the pattern inference is considered as a decision problem referred as to the *consistency problem*.

Ishizaka et al [7] assumed a hypothesis class consisting of all the sets of at most two tree patterns, where a tree pattern is a first order term. They proved the consistency problem for the class to be NP-complete. Thus, in order to overcome this computational hardness, they used the membership oracle, and showed that the problem is decidable polynomially.

Other nice result is due to Kearns and Pitt [9] in probabilistic point of view. They investigated PAC-learnability of k-variable patterns from positive and negative data, and proposed an algorithm for a target pattern to produce a polynomial sized union of patterns under the product distribution.

As was seen in the above, the efficient computability of these problems for pattern languages is closely related to the possibility of efficient machine learning. However, very few problems for the class of patterns are known to be computable efficiently except one-variable patterns. On the other hand, the class of one-variable patterns is attractive, because by the result in [9], the consistency problem for the one-variable patterns is expected to be in inside of NP considered as a proper subclass of NP Thus, there is still room for improvement in their result limited to the one-variable patterns; First, the distribution is enough universal but not arbitrary. Second, it is not required to find a one-variable pattern just fitting for the given examples provided it exists. In the present paper, we focus on tractability of the consistency problem, and show that the above restrictions can be relaxed for the class of one-variable patterns, i.e., the consistency problem is in P.

2 Preliminaries

For each finite set S, the cardinality of S is denoted by $||S||$. An alphabet is a finite set of symbols, denoted by Σ. The free monoid over Σ is denoted by Σ^*, and the set of all non-empty strings is denoted by Σ^+, where $\Sigma^* = \Sigma^+ \cup \{\varepsilon\}$ and ε is the empty string [6].

Let α and β be strings. By $\sharp(\alpha, \beta)$, we denotes the number of occurrences of the β in the α. The length of α is denoted by $|\alpha|$. The i-th symbol of α from the left-to-right order is denoted by $\alpha[i]$. A string $\alpha\alpha$ maybe denoted by $(\alpha)^2$, and similarly, the n-concatenation of α is denoted by $(\alpha)^n$.

Let w be a string of the form $\alpha\beta\gamma$. If the β is an i-th occurrence of it in w from the left-to-right order, then the number $|\alpha| + 1$ is called the position of i-th occurrence of β in w. The α, β and γ are called substrings of w. One of them is called a proper substring if it is not equal to w. The α and γ are respectively said to be a prefix and a suffix of w. In particular, the α is called a proper prefix if $\alpha \neq w$, and a proper suffix is defined similarly.

Let x be a special symbol not belonging in Σ. Every $\pi \in (\Sigma \cup \{x\})^+$ is called a one-variable pattern and the x is referred to as the variable of π. In particular, if $\sharp(\pi, x) \geq 1$, then it is called proper[1]. The class of all one-variable patterns is denoted by Pat.

For any $\pi \in Pat$ and $u \in \Sigma^+$, the expression $\pi[x/u]$ is the string $w \in \Sigma^+$ obtained by replacing all x in π by u. The string u is called a substitution[2] for x of π. For every $\pi \in Pat$, we define the language of π by

$$L(\pi) =_{def} \{w \in \Sigma^+ \mid \exists u \in \Sigma^+,\ w = \pi[x/u]\}.$$

We assume two finite subsets of Σ^+, denoted by S_1 and S_0, where $S_1 \cap S_0 = \emptyset$. A member of S_1 is called a positive example and a member of S_0 is called a negative example. A $\pi \in Pat$ is called consistent with $S_1 \cup S_0$ if $S_1 \subseteq L(\pi)$ and $S_0 \cap L(\pi) = \emptyset$.

Definition 1 Given S_1 and S_0, the consistency problem is to decide whether or not there exists a $\pi \in Pat$ consistent with the $S_1 \cup S_0$.

In [1], Angluin introduced a method for a compact representation of one-variable patterns that generate a same string. Let $w \in \Sigma^+$ and (i, j, k) be a

[1] In general, since $||S_1|| \geq 2$, a consistent π must be proper.

[2] In this study, no erasing for x is assumed, i.e., any substitution is not ε.

triple of nonnegative integers such that $0 \leq i < |w|$, $1 \leq j \leq |w|$, $1 \leq k \leq i+1$ and $j \leq (|w| - i)$. All the (i, j, k) are called *feasible* for w [1].

For each feasible (i, j, k) for a w, let $L_w(i, j, k)$ denote the set of all $\pi \in Pat$ such that $\sum_{a \in \Sigma} \sharp(\pi, a)$ is i, $\sharp(\pi, x)$ is j and the position of the leftmost occurrence of x in π is k.

Every language $L_w(i, j, k)$ can be recognized by a particular finite automaton, called a one-variable pattern automaton [1]. It is well-known that for each $L_w(i, j, k)$, we can compute a one-variable pattern automaton A such that $L(A) = L_w(i, j, k)$.

In general, given n finite automata, it is very hard to compute a finite automaton that defines the language of n-production of them, even if they are all deterministic. On the other hand, let $A_1, A_2, \ldots, A_n$ be one-variable pattern automata. Angluin [1] proved that we can compute a one-variable pattern automaton A' such that $L(A') = \cap_{1 \leq i \leq n} L(A_i)$ in polynomial time in the sum of the size of them, where the size of A_j denotes the number of states of A_j $(1 \leq j \leq n)$.

3 Tractability of consistency problem

Let u be a substring of a string w such that $\sharp(w, u) = r \geq 2$. For each $1 \leq i < j \leq r$, let $p(i)$ and $p(j)$ be the positions of the i-th and j-th occurrences of u, respectively. If $p(j) - p(i) < |u|$, then the suffix of u of length $|u| - (p(j) - p(i))$ is said to be the *overlap* of the i-th and j-th occurrences. We denote an i-th occurrence of u in w by $(u, i)_w$.

The u is said to be a *covering factor* of the w if each $w[i]$ $(1 \leq i \leq |w|)$ is contained in an occurrence of u in w. Suppose that u is a covering factor of w such that $\sharp(w, u) = r \geq 2$. Then, there are the following two cases.

Case 1. There exists an $1 \leq i \leq r - 1$ such that the length of the overlap of $(u, i)_w$ and $(u, i + 1)_w$ is greater than $\lfloor |u|/2 \rfloor$.

Case 2. For any $1 \leq i \leq r - 1$, it is less than or equal to $\lfloor |u|/2 \rfloor$.

Lemma 1 The Case 1 is equivalent to the following conditions.

1. $u = (\alpha\beta)^n\alpha$, where $n \geq 2$.
2. $w = v_1 v_2 \cdots v_m$ such that for each $1 \leq \ell \leq m$, $v_\ell = (\alpha\beta)^{n_\ell}\alpha$ and $n_\ell \geq n$.

3. $r = \sum_{1 \leq \ell \leq m}(n_\ell - n + 1)$.

PROOF. If an overlapping of two occurrences of u in w is greater than $\lfloor |u|/2 \rfloor$, then u must be of the form $(\alpha\beta)^n\alpha$ for a prefix $\alpha\beta$ of u, where $n \geq 2$. Taking such a shortest prefix of u, the condition 1 is satisfied.

Since u is a covering factor of w, we can assume that $w = uv_1uv_2 \cdots uv_{m'}$, where $0 \leq |v_\ell| < |u|$ and $1 \leq \ell \leq m'$. Since $|v_\ell| < |u|$, each v_ℓ must be covered by at most two occurrence of u. Thus, the v_ℓ is of the form $(\beta\alpha)^{i_\ell}\beta$ or $(\beta\alpha)^{i_\ell}(\alpha\beta)^{j_\ell}$. It follows that for each $1 \leq \ell \leq m'$, $uv_\ell u = (\alpha\beta)^{n+i_\ell}\alpha$ or $uv_\ell u = (\alpha\beta)^{n+i_\ell}\alpha(\alpha\beta)^{n+j_\ell}\alpha$. Hence, the condition 2 holds. The condition 1 and 2 follow the condition 3 and the converse direction is clear. □

Lemma 2 The Case 2 is equivalent to the following conditions.

1. $\sharp(uvu, u) \leq 3$, where $|v| \leq |u|$.
2. $w = uv_1uv_2 \cdots uv_mu$ such that for each $1 \leq \ell \leq m$, $|v_\ell| < |u|$ and if $|v_\ell| > 0$, then $\sharp(uv_\ell u, u) = 3$.
3. $r = m' + m + 1$, where $m' = m$ if $\sharp(uu, u) = 3$ and m' is the number of $v_\ell \neq \varepsilon$ if $\sharp(uu, u) = 2$.

PROOF. If there exists a $v \in \Sigma^*$ such that $\sharp(uvu, u) \geq 4$, then an overlap of two occurrences of u in uvu is greater than $\lfloor |u|/2 \rfloor$. It is a contradiction, and similarly, $\sharp(uu, u) = 3$ iff $u = \alpha\alpha$.

Let v_ℓ be the string between $(u, \ell)_w$ and $(u, \ell + 2)_w$. By the assumption, they have no overlap. Since u is a covering factor of w, the v_ℓ must be covered by the $(u, \ell + 1)$ and $|v| \leq |u|$. It follows $\sharp(uv_\ell u, u) = 3$. Thus, the condition 1 and 2 are satisfied.

If $\sharp(uu, u) = 3$, then for each $1 \leq \ell \leq m$, the u and v_ℓ must satisfy $\sharp(uv_\ell u, u) = 3$. If not, we count only the number of $v_\ell \neq \varepsilon$. Hence, we arrive at the condition 3. The converse direction of this proof is obvious. □

Let w be a string and (i, j, k) be a feasible triple for the w. We recall the set $L_w(i, j, k)$ of all $\pi \in Pat$ consistent with w with respect to (i, j, k). Then, w is of the form $w = \gamma_0u\gamma_1w_1\gamma_2w_2 \cdots \gamma_nw_n\gamma_{n+1}$, where $|\gamma_0| = k - 1$, $|u| = (|w| - i)/j$ and u is a covering factor of each w_ℓ $(1 \leq \ell \leq n)$. Thus, each selection of j occurrences not overlapping each other is corresponding to a $\pi \in L_w(i, j, k)$.

Assuming that the (i,j,k) is also feasible for other string w', the set of one-variable patterns consistent with w' with respect to the (i,j,k) is similarly denoted by $L_{w'}(i,j,k)$. Let $L = L_w(i,j,k) \cap L_{w'}(i,j,k)$.

If $L \neq \emptyset$, then we can assume that there exists a unique division of $j = j_1 + j_2 + \cdots + j_n$ such that $w' = \gamma_0 u' \gamma_1 w'_1 \gamma_2 w'_2 \cdots \gamma_n w'_n \gamma_{n+1}$, $|u'| = (|w'| - i)/j$ and for each $1 \leq \ell \leq n$, w'_ℓ contains at least j_ℓ occurrences of u' not overlapping each other. For each $1 \leq \ell \leq n$, let us define the followings.

$$L_{j_\ell}(w_\ell, u) = \{\pi \in Pat \mid \sharp(\pi, x) = j_\ell,\ \pi[x/u] = w_\ell\},$$

$$L_{j_\ell}(w'_\ell, u') = \{\pi \in Pat \mid \sharp(\pi, x) = j_\ell,\ \pi[x/u'] = w'_\ell\} \text{ and}$$

$$L_{j_\ell} = L_{j_\ell}(w_\ell, u) \cap L_{j_\ell}(w'_\ell, u').$$

Since for each $1 \leq \ell < \ell' \leq n$, the selections of $\pi \in L_\ell$ and $\pi' \in L_{\ell'}$ are unconnected each other, it holds that $\pi \in L$ iff $\pi = \gamma_0 x \gamma_1 \pi_1 \gamma_2 \pi_2 \cdots \gamma_n \pi_n \gamma_{n+1}$ and $\pi_\ell \in L_{j_\ell}$ $(1 \leq \ell \leq n)$. Thus, in the discussion below, we can estimate each L_{j_ℓ} individually. For an L_{j_ℓ}, there exist the following two cases.

Case A. The u' is a covering factor of all w'_ℓ.

Case B. There is a w'_ℓ not covered by the u'.

Lemma 3 Let a (w'_ℓ, u') be in Case A. If $||L_{j_\ell}|| \geq 2$, then $L_{j_\ell} = L_{j_\ell}(w_\ell, u)$.

Proof. In this proof, we denote w_ℓ by w and w'_ℓ by w'. There are three combinations of conditions such that (1) (w,u) is in Case 1 and (w',u') is in Case 2; (2) both are in Case 2; and (3) both are in Case 1.

The proof of (1). The (w,u) satisfies that $u = (\alpha\beta)^n\alpha$ and $w = v_1 v_2 \cdots v_m$, where $v_i = (\alpha\beta)^{n_i}\alpha$, $n_i \geq n$ and $1 \leq i \leq n$. The proof is done by induction of the number m for $w = v_1 v_2 \cdots v_m$. Let $m = 1$ such that $w = (\alpha\beta)^{n'}\alpha$, $u = (\alpha\beta)^n\alpha$ and $n' \geq n$.

Assume that there exist distinct $\pi_1, \pi_2 \in L_{j_\ell}$. Let us consider the situation that for any different occurrences $(u,i)_w$ and $(u,j)_w$ such that $(u,i)_w$ is covered by an x of π_1 and $(u,j)_w$ is covered by an x of π_2, the $(u,i)_w$ and $(u,j)_w$ have no overlapping factor. Since there exists a position s such that $\pi_1[s] = x$ and $\pi_2[s] \in \Sigma$, the situation requires at least $u = u'$. It follows the trivial condition $(w,u) = (w',u')$.

Then, we suppose the contrary that some $(u,i)_w$ and $(u,j)_w$ has an overlapping factor. Let $(u,i)_w$ and $(u,j)_w$ begin at the positions s and s' of w, respectively. By the form $u=(\alpha\beta)^n\alpha$, the substring from $w[s]$ to $w[s'-1]$ must be $(\alpha\beta)^d$ and the substring from $w[s+|u|]$ to $w[s'+|u|-1]$ must be $(\beta\alpha)^d$ for some $1 \leq d < n$.

Since they are contained in w', and π_1 and π_2 generate the u', the observation follows that the u' has a prefix $(\alpha\beta)^d$ of at least greater than $\lfloor |u'|/2 \rfloor$ and has a suffix $(\beta\alpha)^d$ of at least greater than $\lfloor |u'|/2 \rfloor$. However, any overlap of two occurrences of u' in w' must not be greater than the half of $|u|$, that is $u'=\alpha\beta\alpha$. Moreover, the condition $\sharp(u'vu',u)=3$ is satisfied only if $v=\beta$.

Each $\pi \in L_{j_\ell}(w,u)$ is of the form $\pi = \beta_0 x \beta_1 x \beta_2 \cdots x \beta_{j_\ell} x \beta_{j_\ell+1}$, where $\beta_0 = (\alpha\beta)^i$, $\beta_{j_\ell+1} = (\beta\alpha)^{i'}$, $(i, i' \geq 0)$ and $\beta_i = (\beta\alpha)^{k_i}\beta$ or ε $(1 \leq i \leq j_\ell)$. Hence, $\pi[x/u'] = w'$, that is $L_{j_\ell} = L_{j_\ell}(w_\ell, u)$. It completes the base step.

In case of $m \geq 2$, since for each $1 \leq i \leq m-1$, there is no overlapping occurrence of u across v_i and v_{i+1}, the number of variables assigned to each v_i is fixed. Then, the result of the base step can be applied to all v_i, independently. Thus, induction step can be proved analogously.

The proof of (2). Assume that there exists $\pi_1, \pi_2 \in L_{j_\ell}$. Similarly to (1), there exist positions s and s' such that $\pi_1[s] = x$ and $\pi_2[s] \in \Sigma$ and $\pi_1[s'] \in \Sigma$ and $\pi_2[s] = x$. Let $d = max\{\lfloor |u|/2 \rfloor, \lfloor |u'|/2 \rfloor\}$. Since u and u' is in Case 2, they must have a same prefix of length at least $d+1$ and a same suffix of length at least $d+1$, that is $u = u' = \alpha\beta\alpha$. Thus, $L_{j_\ell} = L_{j_\ell}(w_\ell, u)$.

The proof of (3). If (w,u) and (w',u) are in Case 1, then it is clear that $L_{j_\ell} \neq \emptyset$ iff $L_{j_\ell} = L_{j_\ell}(w_\ell, u)$. Therefore, we conclude the result. □

In Case B, the occurrences of u' divide w'_ℓ into some h covering factors, and the corresponding partitions are considered for the w_ℓ as follows.

$w_\ell = \gamma_{\ell_1} w_{\ell_1} \gamma_{\ell_2} w_{\ell_2} \cdots \gamma_{\ell_h} w_{\ell_h} \gamma_{\ell_{h+1}}$ and

$w'_\ell = \gamma_{\ell_1} w'_{\ell_1} \gamma_{\ell_2} w'_{\ell_2} \cdots \gamma_{\ell_h} w'_{\ell_h} \gamma_{\ell_{h+1}}$, where

for each $1 \leq i \leq m$, the u is a covering factor of each w_{ℓ_i} and the u' is a covering factor of each w'_{ℓ_i}.

The constant strings $\gamma_{\ell_1}, \ldots, \gamma_{\ell_{h+1}}$ partition w_ℓ and w'_ℓ. Accordingly, the selection of j_ℓ variables is divided into h selections, namely, $j_{\ell_1}, j_{\ell_2}, \ldots, j_{\ell_h}$. The j_{ℓ_i} variables are selected from w_{ℓ_i} and w'_{ℓ_i}. Clearly, the cardinality of L_{j_ℓ} is the product of such possible selections of j_{ℓ_i} variables for all $1 \leq i \leq h$. For the above (w_ℓ, u) and (w'_ℓ, u'), we next show the following lemma.

Lemma 4 Let the (w'_ℓ, u') be in the Case B. Then, $||L_{j_\ell}|| \leq 1$.

PROOF. For each $1 \leq i \leq h$, let $L_{\ell_{j_i}} = L_{\ell_{j_i}}(w_{\ell_{j_i}}, u) \cap L_{\ell_{j_i}}(w'_{\ell_{j_i}}, u')$. It is sufficient to prove that $||L_{\ell_{j_i}}|| \leq 1$. For the simplicity of this proof, we omit the index ℓ from all notations, that is, w_ℓ and w'_ℓ are denoted by w and w', respectively, γ_{ℓ_i} and w_{ℓ_i} are denoted by γ_i and w_i, respectively. In particular, the set $L_{\ell_{j_i}}$ is denoted by L.

Let us take some (w_i, u) and (w'_i, u'). By Lemma 3, we can disregard the case that (w_i, u) and (w'_i, u') are both in Case 2. The remained parts of the proof are (1) (w_i, u) is in Case 1; and (2) (w_i, u) is in Case 2 and (w'_i, u') is in Case 1. If $u = u'$, then u' is a covering factor of w'. Thus, we can assume that $u \neq u'$.

The proof of (1). Since u is a covering factor of w, each γ_i is of the form $\gamma_i = (\beta\alpha)^{n_i}\beta$. Let the (w'_i, u') be in Case 1. The u' and w'_i must be of the form $u' = (\alpha'\beta')^{n'}\alpha'$ and $w'_i = v_1 v_2 \cdots v_m$, where $v_j = (\alpha'\beta')^{n_j}\alpha'$, $n_j \geq n'$ and $1 \leq j \leq m$. If $\alpha'\beta' = \alpha\beta$, then u' is a covering factor of the w'. Thus, we can assume $\alpha' \neq \alpha$ or $\beta' \neq \beta$. Let $\alpha' \neq \alpha$. If $L \neq \emptyset$, then all α' of w'_i must be hidden by variables. For each (w'_i, u'), there exists at most one possibility. The case $\beta' \neq \beta$ is analogous.

Let the (w'_i, u') be in Case 2. As was shown in Lemma 3, if there exists two possibilities for substitutions, then $u' = \alpha\beta\alpha$. Thus, u' is a covering factor of w'. This is a contradiction.

The proof of (2). The (w', u') is of the form $u' = (\alpha'\beta')^{n'}\alpha'$ and $w'_i = v_1 v_2 \cdots v_m$, where $v_j = (\alpha'\beta')^{n_j}\alpha'$, $n_j \geq n'$ and $1 \leq j \leq m$. Similarly, if there exists two possibilities for substitutions, then $u = \alpha'\beta'\alpha'$. Thus, each γ_i $(1 \leq i \leq h)$ must be of the form $\gamma_i = (\beta'\alpha')^{n_i}\beta'$, that is u' is a covering factor of w'. By this contradiction, all the proofs are finished. Hence, for each $1 \leq i \leq h$, there exists at most one possibility of substitutions to generate both (w_i, u) and (w'_i, u'). Therefore, we conclude $||L|| \leq 1$. □

Let S and S' be finite sets of strings such that $S \subseteq S'$ and $||S|| \geq 2$. Let $L = \{\pi \in Pat \mid S \subset L(\pi)\}$ and $L' = \{\pi' \in Pat \mid S' \subset L(\pi')\}$. It is always true that $L' \subseteq L$ [1]. Finally, we show the announced result.

Theorem 1 Given the set $S_1 \cup S_0$ of positive and negative examples, the consistency problem for the class of one-variable patterns is in P.

PROOF. Let (i,j,k) be a feasible triple for all strings in $S_1 \cup S_0$ and let $L(i,j,k)$ denote the set of all $\pi \in Pat$ consistent with S_1 with respect to the (i,j,k). For each $w' \in S_0$, let $L_{w'}(i,j,k) = \{\pi \in L(i,j,k) \mid w' \in L(\pi)\}$.

A string $w \in S_1$ is of the form $w = \gamma_0 u \gamma_1 w_1 \gamma_2 w_2 \cdots \gamma_n w_n \gamma_{n+1}$ such that $|\gamma_0| = k-1$, $|u| = (|w|-i)/j$ and the u is a covering factor of w_ℓ $(1 \leq \ell \leq n)$. Moreover, there exists a unique division of $j = j_1 + j_2 + \cdots + j_n$ such that exactly j_ℓ occurrences of u are selected from w_ℓ as variables. For each $1 \leq \ell \leq n$, let $L_{j_\ell} = \{\pi \in Pat \mid \pi[x/u] = w_\ell,\ \sharp(\pi,x) = j_\ell\}$.

On the other hand, each $w' \in S_0$ is also divided by the (i,j,k) such that $w' = \gamma_0 u' \gamma_1 w'_1 \gamma_2 w'_2 \cdots \gamma_n w'_n \gamma_{n+1}$. Similarly, let $L^{w'}_{j_\ell} = \{\pi \in Pat \mid \pi[x/u'] = w'_\ell,\ \sharp(\pi,x) = j_\ell\}$.

Let $S^\ell_0 = \{w' \in S_0 \mid L_{j_\ell} \neq L^{w'}_{j_\ell}\}$. Moreover, let $\langle S^\ell_0 \rangle = S^\ell_0$ if $||S^\ell_0|| < ||L_{j_\ell}||$, and $\langle S^\ell_0 \rangle = \emptyset$ otherwise. By Lemma 3 and 4, if $L_{j_\ell} \neq L_{j_\ell} \cap L^{w'}_{j_\ell}$, then $||L_{j_\ell} \cap L^{w'}_{j_\ell}|| \leq 1$. Thus, there exists a one-variable pattern consistent with the $S_1 \cup S_0$ iff $\sum_{1 \leq \ell \leq n} \langle S^\ell_0 \rangle = S_0$. Since for each $1 \leq \ell \leq n$, $||S^\ell_0|| \leq ||S_0||$, we can compute S^ℓ_0 and $\langle S^\ell_0 \rangle$ polynomially. The number of all feasible (i,j,k) for $S_1 \cup S_0$ is bounded by a polynomial in the length of a shortest example. Consequently, the consistency problem for the class of one-variable patterns is decidable in polynomial time in the number of examples and in the length of a longest example. □

References

[1] D. Angluin. Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, 21:46-62, 1980.

[2] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319-342, 1988.

[3] H. Arimura, T. Shinohara and S. Otsuki. Finding minimal generalizations for unions of pattern languages and its application to inductive inference from positive data. In *Proc. STACS'94*, LNCS 775, pp. 649-660, 1994. Springer-Verlag.

[4] T. Erlebach, P. Rossmanith, H. Stadtherr, A. Steger and T. Zeugmann. Learning one-variable pattern languages very efficiently on average, in parallel, and by asking queries. In *Proc. 8th International Workshop on Algorithmic Learning Theory*, LNAI 1316, pp. 260-276, Berlin, 1997. Springer-Verlag.

[5] M. R. Garey and D. S. Johnson. Computers and Intractability. W. H. Freeman and company, 1983.

[6] J.E. Hopcroft and J.D. Ullman. Introduction to automata theory, languages, and computation. Addison-Wesley Publ., 1979.

[7] H. Ishizaka, H. Arimura and T. Shinohara. Finding tree patterns consistent with positive and negative examples using queries. In *Proc. 5th International Workshop on Algorithmic Learning Theory*, LNAI 872, pp. 317-332, Berlin, 1994. Springer-Verlag.

[8] T. Jiang, A. Salomaa, K. Salomaa and S. Yu. Inclusion is undecidable for pattern languages. In *Proc. 20th ICALP*, LNCS 700, pp. 301-312, Berlin, 1993. Springer-Verlag.

[9] M. Kearns and L. Pitt. A polynomial-time algorithm for k-variable pattern languages from examples. In *Proc. 2nd Ann. ACM Workshop on Computational Learning Theory*, pp. 57-71, Morgan Kaufmann Publ., San Mateo, 1989.

[10] M. Lothaire. Combinatrics on wards. Addison-Wesley Publ., 1983.

[11] C. H. Papadimitriou. Computational complexity. Addison-Wesley Publ., 1994.

[12] S. Lange and R. Wiehagen. Polynomial-time inference of arbitrary pattern languages. *New Generation Computing*, 8:361-370, 1991.

[13] A. Marron. Learning pattern languages from a single initial example and from queries. In *Proc. 1st Ann. Conference on Computational Learning Theory*, pp. 311-325, 1988.

[14] T. Shinohara and S. Arikawa. Pattern inference. In *Algorithmic Learning for Knowledge-Based Systems*, LNAI 961, pp. 259-291, Berlin, 1995. Springer-Verlag.

[15] T. Shinohara and H. Arimura. Inductive inference of unbounded unions of pattern languages from positive data. In *Proc. 7th International Workshop on Algorithmic Learning Theory*, LNAI 1160, pp. 257-271, Berlin, 1996. Springer-Verlag.

[16] T. Zeugmann. Learning 1-variable pattern languages in linear average time. In *Proc. 11st Ann. Conference on Computational Learning Theory*, 1998, to appear.