Class Preserving Monotonic and Dual Monotonic Language Learning

Steffen Lange [1]
TH Leipzig
FB Mathematik und Informatik
PF 66
O–7030 Leipzig
steffen@informatik.th-leipzig.de

Thomas Zeugmann
TH Darmstadt
Institut für Theoretische Informatik
Alexanderstr. 10
W–6100 Darmstadt
zeugmann@iti.informatik.th-darmstadt.de

Shyam Kapur
Institute for Research in Cognitive Science
University of Pennsylvania
3401 Walnut Street Rm 412C
Philadelphia, PA 19104, USA
skapur@linc.cis.upenn.edu

[1]This research has been partially supported by the German Ministry for Research and Technology (BMFT) under grant no. 01 IW 101.

Abstract

In the present paper strong–monotonic, monotonic, weak–monotonic as well as dual strong–monotonic, dual monotonic and dual weak–monotonic reasoning is studied. We investigate the power of all types of monotonic and dual monotonic inference from positive as well as from positive and negative data in the context of algorithmic language learning theory, thereby obtaining strong hierarchies.

Strong–monotonicity describes the requirement to *only* produce better and better *generalizations*. *Monotonic learning* reflects the demand that for any two guesses the one output later has to be at least as good as the previous one with respect to the language to be learnt. *Weak–monotonicity* is the analogue of *cumulativity* in learning theory.

Dual strong–monotonic learning expresses the concept to generate a chain of descending *specializations* converging in the limit to the target language. *Dual monotonicity* describes the same goal with respect to the target language and *dual weak–monotonic* learning is the dual analogue of cumulativity.

1. Introduction

The process of hypothesizing a general rule from eventually incomplete data (e.g. examples, data obtained by performing experiments a.s.o.) is called inductive inference. In the philosophy of science inductive inference has attracted much attention during the last centuries. Some of the principles developed are very much alive in *algorithmic learning theory*, an emerging science starting with the seminal papers of Solomonoff (1964) and of Gold (1967). Computer scientists widely used their insight into the theory of computability to obtain a better and deeper understanding of processes performing inductive generalizations. In a beautiful paper Angluin and Smith (1987) survey the today state of the art in inductive inference.

The present paper mainly deals with formal language learning, a field in which many interesting and sometimes surprising results have been elaborated within the last decades (cf. e.g. Osherson, Stob and Weinstein (1986), Case (1988), Fulk (1990), Jain and Sharma (1990)). One of the central questions studied so far is whether or not various restrictions on the behavior of a learner do limit the learning capabilities of machines. We shall continue along this line. Before explaining what requirements we want to deal with, let us recall the general situation investigated in language learning. The learner is provided with eventually incomplete information concerning the language to be inferred and has to produce, from time to time, a hypothesis about the phenomenon to be learnt. The information given may contain only *positive examples*, i.e., only strings from the language to be recognized, as well as both *positive and negative examples*, i.e., the learner is fed more and more strings over the underlying alphabet which are classified with respect to their containment to the target language. The space of hypotheses may vary from a particular set of acceptors or grammars to sets of characteristic functions. Moreover, the sequence of hypotheses is required to converge in some specified sense to a hypothesis correctly describing the object to be learnt. In all what follows we exclusively consider learning of recursively enumerable families of uniformly recursive languages. These families are of particular interest with respect to potential applications. As it turned out, it is somtime very important to choose carefully the space of hypotheses. Starting with Angluin's (1980A, 1980B) pioneering papers previous work in the field has mainly dealt with *exact* learning. That means, an enumerable family $\mathcal{L}$ of languages has been required to be inferred with respect to the space of hypotheses $\mathcal{L}$. Instead, we consider *class preserving* language learning, i.e., we demand the hypotheses produced at all to describe a language that is contained in the family of all target languages. This requirement seems to be a very natural one, since any hypothesis not satisfying it, cannot be correct. However, in designing the space of hypotheses we may well change the enumeration as well as the description of all languages from $\mathcal{L}$. The latter approach may considerably increase the learning power, as we shall see (cf. Theorem 3 and Corollary 4).

What we like to present in the sequel is an almost complete investigation of the influence of various monotonicity conditions originally introduced by Jantke (1991A) and Wiehagen (1990) as well as of their dual versions. The main underlying question can be posed as follows: Would it be possible to learn the target language in using more and more information about it and thereby producing *only* better and better generalizations and specializations, respectively? In its strongest interpretation that means we are required to

infer a sequence of hypotheses describing an augmenting (descending) chain of languages, i.e., $L_i \subseteq L_j$ ($L_i \supseteq L_j$) iff L_j is guessed later than L_i (cf. Definition 4 and 5, part (A)).

Dealing with inductive inference of recursive functions Jantke (1991A, 1991B) has shown that strong–monotonicity considerably restricts the inferring power. Starting from the observation that "better" has to be interpreted with respect to the goal of the learning process, i.e., with respect to the language L having to be learnt, Wiehagen (1990, 1991) proposed the following modification: Instead of demanding $L_i \subseteq L_j$ we only require $L_i \cap L \subseteq L_j \cap L$ iff L_j appears later in the sequence of created hypotheses than L_i does (cf. Definition 4 (B)). Intuitively speaking, monotonicity means the following: if the learner's guess at any stage is accurate with regard to the containment of some string s in the target language, then any subsequently produced hypothesis has to contain s as well.

Hence, it is only natural to consider additionally the dual version of monotonicity, i.e., if the learner outputs at some stage a guess correctly excluding a string s from the target language, then any subsequent hypothesis has to behave thus (cf. Definition 5 (B)). The third version of monotonicity, which we call weak–monotonicity and dual weak–monotonicity, respectively, is derived from non–monotonic logics and may be interpreted as the analogue and dual analogue of cumulativity. Consequently, we only require $L_i \subseteq L_j$ and $L_i \supseteq L_j$, respectively, as long as the data obtained after having produced L_i do not contradict L_i (cf. Definition 4 and 5, part (C)).

As it turns out, if the learning process is performed with *positive and negative examples* then weak–monotonicity as well as dual weak–monotonicity do *not restrict* the inferring power as long as the space of hypotheses is suitable chosen (cf. Theorem 14). In case one learns from positive data alone, we show that weak–monotonically working learning devices are exactly as powerful as conservatively working ones. On the other hand, dual weak–monotonic inference from positive data is even more powerful than conservative learning on text. A learning device is said to be *conservative* if and only if it exclusively performs justified mind changes, i.e., it changes its actual hypothesis only in case it is contradicted by new data (cf. Definition 7).

All other notions of monotonicity as well as of dual monotonicity do immediately lead to a severe limitation of the learning power, as we shall show. Finally, in order to avoid confusion it should be mentioned that there is yet another notion of monotonic learning (cf. Osherson, Stob and Weinstein (1986), Fulk (1990)) which is slightly different from ours.

The paper is structured as follows. Section 2 presents preliminaries, i.e., notations, definitions and motivations. In Section 3 we exclusively deal with identification from positive data. The monotonic and dual monotonic inference from positive and negative data is studied in Section 4. Section 5 is devoted to the question whether natural families of languages may be inferred monotonically. Finally, we give a summary and present open problems. All references are listed in Section 7.

2. Preliminaries

By $N = \{1, 2, 3, ...\}$ we denote the set of all natural numbers. In the sequel we assume familiarity with formal language theory (cf. e.g. Bucher and Maurer (1984)). By Σ we

denote any fixed finite alphabet of symbols. Let Σ^* be the free monoid over Σ. The length of a string $s \in \Sigma^*$ is denoted by $|s|$. Any subset $L \subseteq \Sigma^*$ is called a language. By $co-L$ we denote the complement of L, i.e., $co-L = \Sigma^* \setminus L$. Let L be a language and $t = s_1, s_2, s_3, ...$ a sequence of strings from Σ^* such that $range(t) = \{s_k \mid k \in N\} = L$. Then t is said to be a *text* for L or, synonymously, a positive presentation. Furthermore, let $i = (s_1, b_1), (s_2, b_2), ...$ be a sequence of elements of $\Sigma^* \times \{+, -\}$ such that $range(i) = \{s_k \mid k \in N\} = \Sigma^*$, $i^+ = \{s_k \mid (s_k, b_k) = (s_k, +), k \in N\} = L$ and $i^- = \{s_k \mid (s_k, b_k) = (s_k, -), k \in N\} = co-L$. Then we refer to i as an *informant*. If L is classified via an informant then we also say that L is represented by positive and negative data. Moreover, let t, i be a text and an informant, respectively, and let x be a number. Then t_x, i_x denote the initial segment of t and i of length x, respectively, e.g., $i_3 = (s_1, b_1), (s_2, b_2), (s_3, b_3)$. Let t be a text and let $x \in N$. Then we set $t_x^+ = \{s_i \mid i \leq x\}$. Furthermore, by i_x^+ and i_x^- we denote the sets $\{s_k \mid (s_k, +) \in i, k \leq x\}$ and $\{s_k \mid (s_k, -) \in i, k \leq x\}$, respectively.

Following Angluin (1980A) we restrict ourselves to deal exclusively with indexed families of recursive languages defined as follows: A sequence $L_1, L_2, L_3, ...$ is said to be an *indexed family* $\mathcal{L}$ of recursive languages provided all L_j are non–empty and there is a recursive function f such that for all numbers j and all strings $s \in \Sigma^*$ we have

$$f(j,s) = \begin{cases} 1 & , \quad if \quad s \in L_j \\ 0 & , \quad otherwise. \end{cases}$$

As an example we consider the set $\mathcal{L}$ of all context–sensitive languages over Σ. Then $\mathcal{L}$ may be regarded as an indexed family of recursive languages (cf. [6]). In the sequel we often denote an indexed family and its range by the same symbol $\mathcal{L}$. What is meant will be clear from the context.

As in Gold (1967) we define an *inductive inference machine* (abbr. IIM) to be an algorithmic device which works as follows: The IIM takes as its input larger and larger initial segments of a text t (an informant i) and it either requires the next input string, or it first outputs a hypothesis, i.e., a number encoding a certain computer program, and then it requires the next input string (cf. e.g. Angluin (1980A)).

At this point we have to clarify what space of hypotheses we should choose, thereby also specifying the goal of the learning process. Gold (1967) and Wiehagen (1977) pointed out that there is a difference in what can be inferred in dependence on whether we want to synthesize in the limit grammars (i.e., procedures generating languages) or decision procedures, i.e., programs of characteristic functions. Case and Lynes (1982) investigated this phenomenon in detail. As it turns out, IIMs synthesizing grammars can be more powerful than those ones which are requested to output decision procedures. Moreover, several surprising results distinguishing language learning from inductive inference of recursive functions are originated in choosing grammars as space of hypotheses instead of characteristic functions. However, in the context of identification of indexed families both concepts are of equal power. Nevertheless, we decided to require the IIMs to output grammars. This decision has been caused by the fact that there is a big difference between the possible monotonicity requirements. A straightforward adaptation of the approaches made in inductive inference of recursive functions directly yields analogous requirements with respect to the corresponding characteristic functions of the languages to be inferred. On

the other hand, it is only natural to interpret monotonicity with respect to the language to be learnt, i.e., to require containment of languages as described in the introduction. As it turned out, the latter approach increases considerably the power of monotonic and dual monotonic language learning. Furthermore, since we exclusively deal with *class preserving* learning of indexed families $\mathcal{L} = (L_j)_{j \in N}$ of recursive languages we always take as space of hypotheses an enumerable family of grammars $G_1, G_2, G_3, ...$ over the terminal alphabet Σ satisfying $\mathcal{L} = \{L(G_j) \mid j \in N\}$. Moreover, we always require that membership in $L(G_j)$ is uniformly decidable for all $j \in N$ and all strings $s \in \Sigma^*$. As it turns out, it is sometimes very important to choose the space of hypotheses appropriately in order to achieve the desired learning goal. Then the IIM outputs numbers j which we interpret as G_j.

A sequence $(j_x)_{x \in N}$ of numbers is said to be convergent in the limit if and only if there is a number j such that $j_x = j$ for almost all numbers x.

Definition 1. (Gold, 1967) *Let $\mathcal{L}$ be an indexed family of languages, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j \in N}$ be a space of hypotheses. An IIM M $LIM-TXT$ $(LIM-INF)$–identifies L on a text t (an informant i) with respect to $\mathcal{G}$ iff it almost always outputs a hypothesis and the sequence $(M(t_x))_{x \in N}$ $(M(i_x))_{x \in N}$ converges in the limit to a number j such that $L = L(G_j)$.*
Moreover, M $LIM-TXT$ $(LIM-INF)$–identifies L, iff M $LIM-TXT$ $(LIM-INF)$–identifies L on every text (informant) for L. We set:
$LIM-TXT(M) = \{L \in \mathcal{L} \mid M\ LIM-TXT-identifies\ L\}$ and define $LIM-INF(M)$ analogously.
Finally, let $LIM-TXT$ $(LIM-INF)$ denote the collection of all families $\mathcal{L}$ of indexed families of recursive languages for which there is an IIM M such that $\mathcal{L} \subseteq LIM-TXT(M)$ $(\mathcal{L} \subseteq LIM-INF(M))$.

Definition 1 could be easily generalized to arbitrary families of recursively enumerable languages (cf. e.g. Osherson et al (1986)). Nevertheless, we exclusively consider the restricted case defined above, since our motivating examples are all families of recursive languages. Moreover, it may be well conceivable that the weakening of $\mathcal{L} = \{L(G_j) \mid j \in N\}$ to $\mathcal{L} \subseteq \{L(G_j) \mid j \in N\}$ may increase the collection of inferrable indexed families. However, it does not, as the following proposition shows.

Proposition 1. (Lange and Zeugmann, 1992A) *Let $\mathcal{L}$ be an indexed family and let $\mathcal{G} = (G_j)_{j \in N}$ be any space of hypotheses such that $\mathcal{L} \subseteq \{L(G_j) \mid j \in N\}$ and membership in $L(G_j)$ is uniformly decidable. Then we have: If there is an IIM M inferring $\mathcal{L}$ on text (informant) with respect to $\mathcal{G}$, then there is also an IIM $\hat{M}$ that learns $\mathcal{L}$ on text (informant) with respect to $\mathcal{L}$.*

It remained open whether or not Proposition 1 may be generalized to spaces of hypotheses not having a uniformly decidable membership problem.

Note that, in general, it is not decidable whether or not M has already inferred L. The next definition strengthens Definition 1 in that we additionally require decidability of whether or not an IIM has successfully finished its learning task.

Definition 2. (Trakhtenbrot and Barzdin, 1970) *Let $\mathcal{L}$ be an indexed family of languages, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j \in N}$ be a space of hypotheses. An IIM M $FIN-$*

TXT $(FIN-INF)$–infers L on a text t (an informant i) with respect to $\mathcal{G}$ iff it outputs a single and correct hypothesis j, i.e., $L = L(G_j)$, and stops thereafter.

Moreover, M $FIN-TXT$ $(FIN-INF)$–identifies L, iff M $FIN-TXT$ $(FIN-INF)$–identifies L on every text (informant) for L. We set: $FIN-TXT(M) = \{L \in \mathcal{L} \mid M$ $FIN-TXT$–identifies $L\}$ and define $FIN-INF(M)$ analogously.

The resulting identification type is denoted by $FIN-TXT$ and $(FIN-INF)$, respectively.

Lange and Zeugmann (1992C) proved that $ExactFIN-TXT = FIN-TXT$ as well as $ExactFIN-INF = FIN-INF$, i.e., if an indexed family $\mathcal{L}$ is finitely inferrable with respect to some space of hypotheses, then it may always be learnt finitely with respect to $\mathcal{L}$. However, for all modes of identification defined below the situation is totally different.

Next we want to formally define strong–monotonic, monotonic and weak–monotonic inference. In order to do this, first we have to explain what does it mean that an IIM *consistently* identifies a language. Consistent IIM have been introduced by Barzdin (1974). Intuitively speaking, consistency means that the IIM does correctly reflect the part of the language the IIM has already been fed with when it produces a guess.

Definition 3. (Barzdin, 1974) *An IIM M $CONS-TXT$ $(CONS-INF)$–identifies L on a text t (an informant i) iff*

(1) M $LIM-TXT$ $(LIM-INF)$–identifies L on t (on i).

(2) Whenever M on t_x (i_x) produces a hypothesis j_x then $t_x^+ \subseteq L(G_{j_x})$ $(i_x^+ \subseteq L(G_{j_x})$ and $i_x^- \subseteq co-L(G_{j_x}))$.

M $CONS-TXT$ $(CONS-INF)$–identifies L iff M $CONS-TXT$ $(CONS-INF)$–identifies L on every text t (informant i).

By $CONS-TXT(M)$ $(CONS-INF(M))$ we denote the set of all languages which M does $CONS-TXT$ $(CONS-INF)$–identify. $CONS-TXT$ and $CONS-INF$ are analogously defined as above.

The next definition formalizes all the notions of monotonic language learning that we have informally described in the introduction.

Definition 4. (Jantke, 1991A, Wiehagen, 1991) *An IIM M is said to identify a language L from text (informant)*

(A) strong–monotonically

(B) monotonically

(C) weak–monotonically

iff
M $LIM-TXT$ $(LIM-INF)$–identifies L and for any text t (informant i) of L as well as for any two consecutive hypotheses j_x, j_{x+k} which M has produced when fed t_x and t_{x+k} (i_x and i_{x+k}), for some $k \geq 1, k \in N$, the following conditions are satisfied:

(A) $L(G_{j_x}) \subseteq L(G_{j_{x+k}})$

(B) $L(G_{j_x}) \cap L \subseteq L(G_{j_{x+k}}) \cap L$

(C) *if* $t_{x+k} \subseteq L(G_{j_x})$, *then* $L(G_{j_x}) \subseteq L(G_{j_{x+k}})$ *(if* $i^+_{x+k} \subseteq L(G_{j_x})$ *and* $i^-_{x+k} \subseteq co-L(G_{j_x})$, *then* $L(G_{j_x}) \subseteq L(G_{j_{x+k}})$*).*

Remark: (C) in particular means that M has to work strong–monotonically as long as its guess j is consistent with the data fed to M after M has output j.

By $SMON-TXT$, $SMON-INF$, $MON-TXT$, $MON-INF$, $WMON-TXT$, and $WMON-INF$ we denote the family of all thoses sets $\mathcal{L}$ of languages for which there is an IIM inferring it strong–monotonically, monotonically, and weak–monotonically from text and informant, respectively.

This seems to be a good place to give an example showing what inferring power our approach really implies. Let $L \subseteq \Sigma^*$ be any arbitrarily fixed *infinite* context–sensitive language. By $\mathcal{L}_{fin}$ we denote the set of all finite languages over Σ. Then we set $\mathcal{L}_{finvar} = \{L \cup L_{fin} \mid L_{fin} \in \mathcal{L}_{fin}\}$. In case we would require the IIMs to output programs of characteristing functions one immediately obtains from Jantke (1991A) that, even on informant, $\mathcal{L}$ cannot be learnt strong–monotonically. On the other hand, in our setting $\mathcal{L}_{finvar}$ is strong–monotonically learnable, even on *text*.

The IIM M performing the inference process initially outputs a grammar G of L. Let j_x be $M's$ guess after having received t_x. If $t^+_x \subset t^+_{x+1}$ then M tests whether the new string s belongs to $L(G_{j_x})$. In case it does, we set $j_{x+1} = j_x$. Otherwise M outputs a canonical number j_{x+1} of the grammar $G_{j_x} \cup \{\sigma \rightarrow s\}$ where σ denotes the distinguished non–terminal of G_{j_x} from which every derivation starts. Since for any L' in $\mathcal{L}_{finvar}$ there is a $L_{fin} \in \mathcal{L}_{fin}$ such that $L' = L \cup L_{fin}$ the "else" case can happen at most finitely often. Hence, M converges and by construction M does work strong–monotonically.

Next to we define dual monotonic language learning.

Definition 5. *An IIM M is said to identify a language L from text (informant)*

(A) dual strong–monotonically

(B) dual monotonically

(C) dual weak–monotonically

iff

M $LIM-TXT$ $(LIM-INF)$–identifies L and for any text t (informant i) of L as well as for any two consecutive hypotheses j_x, j_{x+k} which M has produced when fed t_x and t_{x+k} (i_x and i_{x+k}), for some $k \geq 1, k \in N$, the following conditions are satisfied:

(A) $co-L(G_{j_x}) \subseteq co-L(G_{j_{x+k}})$

(B) $co-L(G_{j_x}) \cap co-L \subseteq co-L(G_{j_{x+k}}) \cap co-L$

(C) if $t_{x+k} \subseteq L(G_{j_x})$, *then* $co - L(G_{j_x}) \subseteq co - L(G_{j_{x+k}})$ *(if* $i^+_{x+k} \subseteq L(G_{j_x})$ *and* $i^-_{x+k} \subseteq co - L(G_{j_x})$, *then* $co - L(G_{j_x}) \subseteq co - L(G_{j_{x+k}})$*).*

By $SMON^d - TXT$, $SMON^d - INF$, $MON^d - TXT$, $MON^d - INF$, $WMON^d - TXT$, and $WMON^d - INF$ we denote the collections of all those indexed families $\mathcal{L}$ of languages for which there is an IIM identifying it dual strong–monotonically, dual monotonically and dual weak–monotonically from text and informant, respectively.

Note that the notion of monotonicity and of dual monotonicity are truly duals of *each other*. Next to, we combine the monotonicity constraints from Definition 4 and 5. As we shall see, this might help to gain a better understanding of the relationships between monotonic inference of languages and other well–known types of language learning.

Definition 6. *Let* $SMON^{\&} - TXT$ *(*$SMON^{\&} - INF$*) denote the class of indexed families learnable by an IIM that works strong–monotonically as well as dual strong–monotonically. The identification types* $MON^{\&} - TXT$, $MON^{\&} - INF$, $WMON^{\&} - TXT$ *and* $WMON^{\&} - INF$ *are defined analogously.*

Finally in this section we define *conservatively* working IIMs.

Definition 7. (Angluin ,1980A)
An IIM M CONSERVATIVE–TXT (CONSERVATIVE–INF)–identifies L on text t (on informant i), iff for every text t (informant i) the following conditions are satisfied:

(1) $L \in LIM - TXT(M)$ $(L \in LIM - INF(M))$

(2) If M on input t_x *makes the guess* j_x *and then makes the guess* $j_{x+k} \neq j_x$ *at some subsequent step, then* $L(G_{j_x})$ *must fail to contain some string from* t_{x+k} *(*$L(G_{j_x})$ *must fail either to contain some string* $s \in i^+_{x+k}$ *or it generates some string* $s \in i^-_{x+k}$*).*

CONSERVATIVE–TXT(M), CONSERVATIVE–INF(M) as well as CONSERVATIVE–TXT and CONSERVATIVE–INF are defined in an analogous manner as above.

Intuitively speaking, a conservatively working IIM performs *exclusively* justified mind changes. In particular, it is never allowed to output a guess generating a proper superset of the target language.

Sometimes we want to combine some postulates, e.g. strong–monotonicity and consistency. That is denoted by e.g. $SMON - CONS - TXT$ or $SMON - CONS - INF$ and means that the particular IIM which has to perform the inferring process is required to work strong–monotonically *and* consistently on every text and informant, respectively.

Moreover, whenever appropriate we use the prefix *Exact* to denote the exact version of the inference mode under consideration. For example, $ExactWMON - TXT$ denotes exact weak–monotonic language learning from positive data.

In the next section we present results concerning text identification. Note that in all what follows $\subset$ denotes proper set inclusion and # denotes incomparability of sets.

3. Monotonic Inference on Text

Our first theorem actually shows that there is a strong hierarchy between the different notions of monotonicity.

Theorem 1. $SMON - TXT \subset MON - TXT \subset WMON - TXT$

Proof. $SMON - TXT \subseteq MON - TXT$ is an immediate consequence of Definition 4. The part $MON - TXT \setminus SMON - TXT \neq \emptyset$ is proved via the following indexed family of languages:
Let $(i_1, ..., i_k)$ be any collection of pairwise distinct numbers. We define $L_\emptyset = \{a\}^*$ and set $L_{i_1,...,i_k} = (\{a\}^* \setminus \{a^{i_1}, ..., a^{i_k}\}) \cup \{b^{i_1}, ..., b^{i_k}\}$. Let $\mathcal{L}_{mon}$ be the family of all languages $L_\emptyset$, $L_{i_1,...,i_k}$, where $(i_1, ..., i_k) \in N^k, k \in N, k \geq 1, i_j \neq i_l$ if $j \neq l$ for all $j, l \in \{1, ..., k\}$.

First we present an IIM M inferring $\mathcal{L}_{mon}$ monotonically on text. Let t be any fixed text. As long as the part of t fed to M does only contain strings over the alphabet $\{a\}$ the IIM M outputs a canonical grammar for $\{a\}^*$. In case the first string b^{l_1} appears in the text, M outputs a canonical grammar for L_{l_1}. This hypothesis remains unchanged until the next string b^{l_2} appears in the text ($l_1 \neq l_2$). Then M's new output is a canonical grammar for $L_{l_1 l_2}$, a.s.o.

Now suppose $L \in \mathcal{L}_{mon}$. If $L = \{a\}^*$, then M trivially infers L monotonically. Now let $L \neq \{a\}^*$. We have to show that M identifies L monotonically on text. Since $L \in \mathcal{L}_{mon}$ there are a number k and pairwise distinct numbers $i_1, ..., i_k$ such that $L = L_{i_1,...,i_k}$. Moreover, on every text t M eventually outputs $L_{l_1,...,l_k}$ after having fed $b^{l_1}, ..., b^{l_k}$ sometimes, where $(l_1, ..., l_k)$ is a permutation of $(i_1, ..., i_k)$. Consequently, $L_{l_1,...,l_k} = L_{i_1,...,i_k}$. Hence M infers L. Due to the construction of M the IIM M performs at most $k + 1$ mind changes. Therefore, we have to calculate $\{a\}^* \cap L = L_{i_1,...,i_k} \setminus \{b^{i_1}, ..., b^{i_k}\}$, $L_{l_1} \cap L = L_{i_1,...,i_k} \setminus \{b^{i_1}, ..., b^{i_k} \mid i_j \neq l_1,\ 1 \leq j \leq k\}$, $L_{l_1,...,l_k} \cap L = L_{i_1,...,i_k}$. Hence, if $j_1, ..., j_{k+1}$ are M's outputs, then we have $L(G_{j_1}) \cap L \subseteq L(G_{j_2}) \cap L \subseteq ... \subseteq L(G_{j_{k+1}}) \cap L$, i.e., M works monotonically on text. This proves the first part.

It remains to show that there is no IIM identifying $\mathcal{L}_{mon}$ strong–monotonically on text. However, we shall prove the slightly stronger result that $\mathcal{L}_{mon} \notin SMON - INF$.

Suppose the converse, i.e., assume there is an IIM M such that $\mathcal{L}_{mon} \in SMON - INF(M)$. Let $l_1 > 0$ be arbitrarily fixed and let i be any informant over $\Sigma = \{a, b\}$ for L_{l_1}. Then there must be an x such that M after having fed i_x outputs a canonical grammar for L_{l_1}, since otherwise M cannot infer L_{l_1}. Now, let l_2 be any fixed number satisfying $l_2 > max\{|s| \mid s \in i_x\}$. Then consider i_x and $i_x, (b^{l_2}, +)$, two initial segments of an informant for $L_{l_1 l_2}$. Due to our choice of i_x, M first outputs j_x such that $L(G_{j_x}) = L_{l_1}$ and afterwards, on any large enough extension of $i_x \cup \{(b^{l_2}, +)\}$ that is an initial segment of an informant for $L_{l_1 l_2}$, it has to perform at least one mind change. Otherwise M cannot infer $L_{l_1 l_2}$. On the other hand, $L_{l_1} \not\subseteq L_{l_1 l_2}$. Thus M has to perform at least one mind change contradicting the strong–monotonicity condition. This completes the proof of $SMON - TXT \subset MON - TXT$.

Next we show $MON - TXT \subseteq WMON - TXT$. The problem we have to deal with is caused by the fact that any monotonically working IIM may eventually output an overgeneralized hypothesis j, i.e., $L \subset L(G_j)$, while a weakly-monotonically working IIM

may not. The proof is done using the following claim:

Claim. Let $\mathcal{L}$ be any indexed family, and let M be any monotonically working IIM inferring $\mathcal{L}$ with respect to some space $(G_j)_{j\in N}$ of hypotheses. Let $L \in \mathcal{L}$ and let t be any fixed text for L. If M on t produces a guess j such that $L \subset L(G_j)$, then $L(G_j) \notin \mathcal{L}$.

Suppose the converse, i.e., there is an index x such that $M(t_x) = j$, $L \subset L(G_j)$ and $L(G_j) \in \mathcal{L}$. Then t_x is also an initial segment of a text for $L(G_j)$. On the other hand, t_x possesses an enlargement t_{x+k} such that $t^+_{x+k} \subseteq L$ and $j_{x+k} := M(t_{x+k})$ satisfies $L(G_{j_{x+k}}) = L$. Again, t_{x+k} is an initial segment of a text for $L(G_j)$. Consequently, there should be an enlargement of t_{x+k}, say t_{x+k+z}, with strings from $L(G_j)$ such that $j_{x+k+z} := M(t_{x+k+z})$ and $L(G_{j_{x+k+z}}) = L(G_j)$.
Since

$$L(G_j) \supset L = L(G_{j_{x+k}}) \cap L(G_j) \subset L(G_{j_{x+k+z}}) \cap L(G_j) = L(G_{j_{x+k+z}})$$

we have found a text t for $L(G_j)$ on which M does not identify $L(G_j)$ monotonically, a contradiction. This proves the claim.

On the other hand, since we have required $\mathcal{L} = \{L(G_j) \mid j \in N\}$ for any space of hypotheses $(G_j)_{j\in N}$ the wanted inclusion follows immediately by the claim made above.

The remaining part, i.e., $WMON - TXT \setminus MON - TXT \neq \emptyset$ can be proved using the following indexed family $\mathcal{L}_{wmon}$:
Let $L_1 := \{a\}^*$ and for $i > 1$ set $L_i := \{a^z \mid z < i\} \cup \{b^z \mid z \geq i\}$ as well as $L_{i,j} := \{a^z \mid z < i\} \cup \{b^z \mid i \leq z < j\} \cup \{a^z \mid z \geq j\}$ for all $i, j \in N$ with $i < j$, and $i, j > 1$. $\mathcal{L}_{wmon}$ is defined to be the collection of all $L_i, L_{i,j}$. For the demonstration of $\mathcal{L}_{wmon} \in WMON{-}TXT \setminus MON{-}TXT$ the reader is referred to the proof of Theorem 17, Claim C and D, where the slightly stronger result $\mathcal{L}_{wmon} \in WMON{-}TXT \setminus MON{-}INF$ is shown.

q.e.d.

Moreover, $SMON - TXT$ and $MON - TXT$ may already be separated on sets of regular languages. At this point it is only natural to ask what are lower and upper bounds of this hierarchy. The answer is given by the next theorem.

Theorem 2.

(1) $FIN - TXT \subset SMON - TXT$

(2) $WMON - TXT = CONSERVATIVE\text{–}TXT$

Proof. By our definition, if $\mathcal{L} \in FIN - TXT(M)$, then M outputs a single guess. Hence $FIN - TXT \subseteq SMON - TXT$.

The remaining part $SMON - TXT \setminus FIN - TXT \neq \emptyset$ can be directly obtained by proving $\mathcal{L}_{finvar} \notin FIN - TXT$, where $\mathcal{L}_{finvar} = \{L \cup L_{fin} \mid L_{fin} \in \mathcal{L}_{fin}\}$, i.e., the family defined before presenting Definition 5.

Looking at assertion (2) we directly see that $CONSERVATIVE\text{–}TXT \subseteq WMON - TXT$, since a conservatively working IIM only performs justified mind changes.
Now suppose $\mathcal{L} \in WMON - TXT(M')$. We have to construct an IIM M such that $\mathcal{L} \in CONSERVATIVE\text{–}TXT(M)$. M is easily obtained from M' by simply adding a

consistency test, i.e., let $j_x := M(t_x)$ and $s \in t^+_{x+k-1}$ for some $k = 1, 2, \ldots$. Then test whether or not $s \in L(G_{j_x})$ for all s. In case it is, M repeats j_x as its hypothesis. Otherwise, it outputs $M'(t_{x+k})$.

q.e.d.

Next we ask whether or not $LIM - TXT$ and $WMON - TXT$ may be separated. For exact learning, Angluin (1980A) proved conservatively working learning devices to be less powerful than those ones allowed to output overgeneralized hypotheses. However, her proof technique does not seem applicable in the class preserving case. Therefore, we conjectured that exact learning may decrease the learning power. As our next theorem impressively shows, this conjecture is really true.

Theorem 3. $SMON - TXT \setminus ExactWMON - TXT \neq \emptyset$

Proof. Fix any Gödel numbering φ_1, φ_2, φ_3, ... of the partial recursive functions over N. As usual, we use the following abbreviations. By $\varphi_k(k) \downarrow$ we denote that $\varphi_k(k)$ is defined. If $\varphi_k(k)$ is not defined, we write $\varphi_k(k) \uparrow$. Furthermore, we use $\varphi_k^{(j)}(k) \downarrow$, if $\varphi_k(k)$ can be computed within at most j steps. Otherwise, we write $\varphi_k^{(j)}(k) \uparrow$. By $c : N \times N \to N$ we denotes Cantor's pairing function.

The desired indexed family is defined as follows. For all $k, j \in N$:

$L_{c(k,1)} = \{a^k b^n \mid n \in N\}$, and for $j > 1$ let

$$L_{c(k,j)} = \begin{cases} L_{c(k,1)} & , \; if \; \varphi_k^{(j)}(k) \uparrow \\ \{a^k b^m \mid m \leq z\} & , \; if \; z = \mu n \{n \mid n \leq j \; , \; \varphi_k^{(n)}(k) \downarrow\} \end{cases}$$

Obviously, $\mathcal{L}$ is an indexed family of recursive languages. The definition above has the following consequences. (1) For any k, $j \in N$, if $a^k b^{j+1} \notin L_{c(k,j)}$, then $L_{c(k,j)}$ is a finite language. Hence, it is easy to decide whether or not $L_{c(k,j)}$ is finite. This will help to prove Claim A. (2) For any k, $j \in N$, $a^k b^j \in L_{c(k,j)}$ if and only if $a^k b^j \in L_{c(k,j+r)}$, for any $r \in N$. The latter observation is used in proving Claim B.

Claim A. $\mathcal{L} \notin ExactWMON - TXT$

Suppose the converse, i.e., $\mathcal{L} \in ExactWMON - TXT$. Without loss of generality, we may assume that there is an IIM M which conservatively and consistently infers $\mathcal{L}$ w.r.t. the space of hypotheses $\mathcal{L}$ (cf. Theorem 9). We shall show that M decides the halting problem for φ, i.e., M may be used for deciding whether or not $\varphi_k(k)$ is defined. Let $k \in N$, and let t be the canonical text for $L_{c(k,1)}$. Since M in particular infers L on t, there is an minimal z such that M on input t_z computes a hypothesis. Moreover, because M works consistently, we obtain $M(t_z) = c(k, j)$ for some $j \in N$. First of all, we have to test whether or not $a^k b^{j+1} \in L_{c(k,j)}$. This test can be performed effectively.

Case 1. $a^k b^{j+1} \notin L_{c(k,j)}$
Then $L_{c(k,j)}$ is finite. Due to our construction, we obtain that $\varphi_k(k) \downarrow$.

Case 2. $a^k b^{j+1} \in L_{c(k,j)}$
By definition of $\mathcal{L}$ we get that $L_{c(k,j)}$ is infinite, i.e., $L_{c(k,j)} = L_{c(k,1)}$. For deciding whether or not $\varphi_k(k)$ is defined, we have to compare j and z.

Subcase 2.1. $j \geq z$
We claim that $\varphi_k(k)\uparrow$. Suppose the converse. Then, due to our construction, there has to be an $n > j$ such that $L_{c(k,n)}$ is finite. Since $n > j \geq z$, t_z is an initial segment of the canonical text for $L_{c(k,n)}$. Hence, M has produced an overgeneralized hypothesis when inferring $L_{c(k,n)}$ on its canonical text. This contradicts our assumption that M infers $\mathcal{L}$ exactly and conservatively.

Subcase 2.2. $j < z$
Now, one has to test whether or not $a^k b^{z+1} \in L_{c(k,z)}$. If not, $L_{c(k,z)}$ is finite. Hence, $\varphi_k(k)\downarrow$. If $a^k b^{z+1} \in L_{c(k,z)}$, then $L_{c(k,n)} = L_{c(k,1)}$. Now, one may easily conclude that $\varphi_k(k)\uparrow$. Assume the opposite. Then there has to be an $n > z$ such that $L_{c(k,n)}$ is finite. Since $n > z$, t_z is again an initial segment of the canonical text for $L_{c(k,n)}$ which yields an contradiction as in Subcase 2.1.

Hence, there is indeed no IIM at all which weak-monotonically infers $\mathcal{L}$ w.r.t. the space of hypotheses $\mathcal{L}$.

Claim B. $\mathcal{L} \in SMON - TXT$
We define an indexed family $\hat{\mathcal{L}} = (\hat{L}_j)_{j \in N}$ such that $range(\hat{\mathcal{L}}) = range(\mathcal{L})$. Afterwards, we show that there is an IIM M which weak-monotonically infers $\mathcal{L}$ w.r.t. the space of hypotheses $\hat{\mathcal{L}}$. Let $k \in N$.

$$\hat{L}_{2k-1} = L_{c(k,1)}$$

$$\hat{L}_{2k} = \bigcap_{j \in N} L_{c(k,j)}$$

First of all, we show that $\hat{\mathcal{L}}$ is an indexed family satisfying the announced properties. Due to our construction, we may immediately conlude that $range(\hat{\mathcal{L}}) = range(\mathcal{L})$. Moreover, it is easy to verify that, for any number k and any string s, we have $s \in \hat{L}_{2k}$ if and only if there is an j such that $s = a^k b^j$ as well as $s \in L_{c(k,j)}$. Since $\mathcal{L}$ is an indexed family, it is not hard to prove that $\hat{\mathcal{L}}$ is an indexed family, too. We omit the details.

It remains to show that there is an IIM M which strong-monotonically infers $\mathcal{L}$ w.r.t. the space of hypotheses $\hat{\mathcal{L}}$. Assume any $L \in \mathcal{L}$, any text t for L, and any $x \in N$. We set:

$$M(t_x) = \begin{cases} 2k & , \; if \; t_x^+ \subseteq \hat{L}_{2k} \\ 2k-1 & , \; otherwise \end{cases}$$

Since $\hat{\mathcal{L}}$ is an indexed family, M is indeed an IIM. It is easy to recognize that M performs at most one mind change. If L is a finite language, then M outputs in every step a correct hypothesis. If L is an infinite language, M may compute at the beginning a wrong hypothesis j. But in this case, $\hat{L}_j$ is finite and, therefore, there has to be an y such that $t_y^+ \not\subseteq \hat{L}_j$. Due to our definition, M changes its mind to the correct hypothesis $j-1$ which will be repeated in every subsequent step. Moreover, it follows $\hat{L}_j \subseteq \hat{L}_{j-1}$. Hence, M strong-monotonically infers L on text t.

q.e.d.

Theorem 3 directly implies that exact strong–monotonic, exact monotonic and exact weak–monotonic learning are all less powerful than the corresponding class preserving modes of inference. Moreover, in refining the proof technique developed above we obtained the wanted separation of class preserving conservative language learning from positive data and language learning in the limit on text. However, we even proved a slightly stronger result. The following theorem sheds additionally light on the power of dual monotonic inference from positve data.

Theorem 4. $MON^d - TXT \setminus WMON - TXT \neq \emptyset$

Proof. We use the same notation as in the proof of Theorem 3. The desired indexed family is defined as follows. For all $k, j \in N$:

$L_{c(k,1)} = \{a^k b^n \mid n \in N\}$, and for $j > 1$ we define:

if $a^k b^j \in L_{c(k,j-1)}$, then

$$L_{c(k,j)} = \begin{cases} L_{c(k,1)} & , \; if \; \varphi_k^{(j)}(k) \uparrow \\ \{a^k b^m \mid m \leq z\} & , \; if \; z = \mu n\{n \mid n \leq j \; , \; \varphi_k^{(n)}(k) \downarrow\} \end{cases}$$

else

$$L_{c(k,j)} = \begin{cases} \{a^k b^m \mid m \leq y-1\} & , \; if \; y = max\{n \mid a^k b^n \in L_{c(k,j-1)}\} \; and \; y \neq 1 \\ \{a^k b\} & , \; otherwise \end{cases}$$

Obviously, $\mathcal{L}$ is an indexed family of recursive languages.

Claim A. $\mathcal{L} \notin WMON - TXT$

Suppose the converse, i.e. $\mathcal{L} \in WMON - TXT$. Without loss of generality, we may assume that there is an IIM M which conservatively as well as consistently infers $\mathcal{L}$ w.r.t. any appropriate space of hypotheses $\mathcal{G}$ (cf. Theorem 9). We shall show that M decides a restricted version of the halting problem for φ, i.e., M may be used for deciding whether or not $\varphi_k(k)$ is defined for every $k \in N$. Let $k \in N$, and let t be the canonical text for $L_{c(k,1)}$. Since M in particular infers $L_{c(k,1)}$ on t, there is an minimal z such that M on input t_z computes a hypothesis. Moreover, because M works consistently, we obtain $M(t_z) = j$ with $t_z^+ \subseteq L(G_j)$ for any $j \in N$. First of all, we have to test whether or not $a^k b^{z+1} \in L(G_j)$. Since $\mathcal{G}$ is appropriately chosen, this test can be effectively performed.

Case 1. $a^k b^{z+1} \notin L(G_j)$
Since $range(\mathcal{L}) = L(\mathcal{G})$, the language $L(G_j)$ is finite. Due to our construction, we may conlude that $\varphi_k(k)$ is defined.

Case 2. $a^k b^{z+1} \in L_{c(k,j)}$
Then, we have to test whether or not $\varphi_k^{(y)}(k) \downarrow$ for $y = 1, ..., z$. If $\varphi_k^{(y)}(k)$ may be computed in at most y steps for some y, then $\varphi_k(k)$ is defined. Otherwise, we may conlude that $\varphi(k) \uparrow$, i.e., $\varphi(k)$ is undefined. Suppose the converse, i.e., $\varphi_k(k)$ is defined. Hence, $\varphi_k(k)$ may be computed in at most y steps for some $y > z$. Since $range(\mathcal{L}) = L(\mathcal{G})$, we can immediately conclude from our definition of $\mathcal{L}$ that there has to be an index $n \in N$ such

that $L(G_n) = t_z^+$. Obviously, t_z is an initial segment of a text t for $L(G_n)$. Since $M(t_z) = j$ as well as $L(G_n) \cup \{a^k b^{z+1}\} \subseteq L(G_j)$, M has produced an overgeneralized hypothesis when inferring $L(G_n)$ on t. Hence, M does not work weak-monotonically. This contradicts our assumption. Therefore, $\varphi(k)$ is undefined.

Consequently, M may be used to decide whether or not $\varphi_k(k)$ is defined for every $k \in N$. Since this version of the halting problem is undecidable, this contradicts our assumption that there is an IIM M at all which conservatively as well as consistently infers $\mathcal{L}$ w.r.t. any appropriate space of hypotheses.

Claim B. $\mathcal{L} \in MON^d - TXT$

We show that there is an IIM M which infers $\mathcal{L}$ w.r.t. the space of hypotheses $\mathcal{L}$. Let $L \in \mathcal{L}$, let t be any text for L, and let $x \in N$. Due to our construction, there has to be a k such that $t_x^+ \subseteq L_{c(k,1)}$. Moreover, the first string in t determines k. We set:

$M(t_x) =$ "Test for all $j = 1, ..., x$, whether or not $a^k b^x \notin L_{c(k,j)}$. If at least one j fulfils this test, then goto (A). Otherwise, goto (B).

(A) If $card(t_x^+) = 1$, then compute the minimal y with $a^k b^2 \notin L_{c(k,y)}$, and output $c(k,y)$. Otherwise, compute $z = max\{n \mid n \leq x,\ t_x^+ \subseteq L_{c(k,n)}\}$. Output $c(k,z)$.

(B) Output $c(k,1)$".

Since $\mathcal{L}$ is an indexed family, M is indeed an IIM. It remains to show that M infers L dual monotonically on text t. We distinguish between the following cases.

Case 1: $\varphi_k(k) \uparrow$
Since $\varphi_k(k)$ is undefined, we have $L_{c(k,j)} = L_{c(k,1)} = L$ for all $j \in N$. Hence, M will always produce its hypothesis as defined in Stage (B). Therefore, M outputs in every step the same correct hypothesis $c(k,1)$, and obviously, it works dual monotonically.

Case 2: $\varphi_k(k) \downarrow$
Since $\varphi_k(k)$ is defined, there has to be an $y \in N$ such that $\varphi_k(k)$ may be computed in exact y steps. Due to our construction, we have to consider the following subcases.

Subcase 2.1: $card(L) \leq y$
Hence, after fed with t_y the IIM M reaches Stage (A), and for all $z < y$ M outputs $c(k,1)$. Moreover, there exists $z > y$ such that $t_z^+ = L$. It is easy to verify that $M(t_z) = c(k, y+r)$ for some $r \in N$ with $L_{c(k,y+r)} = L$. Furthermore, this hypothesis will be repeated in any subsequent step. Therefore, M infers L. Finally, M works dual monotonically, since $L_{c(k,1)} \cup L \supseteq L_{c(k,y)} \cup L \supseteq ... \supseteq L_{c(k,y+r)} \cup L$. To see this, remember that $L_{c(k,1)} \supset L$ as well as $L_{c(k,y+r)} \subseteq L$ for any $r \in N$. Thus, applying de Morgan's rules it immediately follows that M works indeed dual monotonically.

Subcase 2.2: $card(L) = \infty$
Hence, $L = L_{c(k,1)}$. Obviously, there has to be an $z > y$ such that $t_z^+ \setminus L_{c(k,y)} \neq \emptyset$. Since all finite languages in $\mathcal{L}$ are subsets of $L_{c(k,y)}$, we obtain $t_z^+ \setminus L_{c(k,n)} \neq \emptyset$ for all $n > y$. Since $L_{c(k,y-1)} = L_{c(k,1)} = L$, we have $M(t_z) = c(k, y-1)$. Again, this hypothesis will be repeated in any subsequent step. Finally, M works dual monotonically, since $L \supseteq L_{c(k,j)}$ for any produced hypothesis $c(k,j)$.

Hence, M infers L dual monotonically on text t.

q.e.d.

As an immediate consequence of Theorem 4 we directly obtain the announced separation of class preserving weak–monotonic language learning and language learning in the limit.

Corollary 5. $WMON - TXT \subset LIM - TXT$

Next we ask whether or not analogues of Theorem 1 and 2 are valid for dual monotonic inference. The answer is given by the next theorems.

Theorem 6.

(1) $FIN - TXT = SMON^d - TXT \subset MON^d - TXT \subset WMON^d - TXT$

(2) $CONSERVATIVE\text{–}TXT \subset WMON^d - TXT$

Proof. Trivially, $FIN - TXT \subseteq SMON^d - TXT$. Further, if an IIM inferring any $\mathcal{L}$ dual strong–monotonically shrinks its guess, the previous guess cannot be learnt dual strong–monotonically from text. Hence, the IIM can output only equivalent hypotheses, i.e., if it outputs j and subsequently k, then $L(G_j) = L(G_k)$. Consequently, M's first guess has to be a correct one.

We proceed in showing $FIN - TXT \subset MON^d - TXT$. Since the inclusion is obvious, it remains to prove $MON^d - TXT \setminus FIN - TXT \neq \emptyset$. However, this is an immediate consequence of Theorem 4.

In order to finish the proof of assertion (1) we continue in showing $MON^d - TXT \subseteq WMON^d - TXT$. Consider a machine M that learns an indexed family $\mathcal{L}$ dual monotonically with respect to some space $(G_j)_{j \in N}$ of hypotheses. Suppose, M does not learn dual weak–monotonically as well. Consequently, M has to change its guess, say k, at some stage on consistent data to j such that $co - L(G_k) \not\subseteq co - L(G_j)$. If the presentation is now extended as in the proof of Theorem 1 to a text for $L(G_k)$, the machine cannot revert guessing $L(G_k)$ without violating dual monotonicity. We omit the details. The proper inclusion is an immediate consequence of assertion (2) of this theorem as well as of assertion (2) of Theorem 7 below, since $MON - TXT \subset CONSERVATIVE\text{–}TXT$ by Theorem 1.

Next we prove assertion (2). Since $CONSERVATIVE\text{–}TXT \subseteq WMON^d - TXT$ is obvious, it suffices to show $WMON^d - TXT \setminus CONSERVATIVE\text{-}TXT \neq \emptyset$. Again, the latter statement directly follows from Theorem 4. This proves the theorem.

q.e.d.

The next theorem offers some deeper insight into the inference capabilities of dual monotonically working machines in deriving lower and upper bounds as well as in comparing it with monotonic inference on text.

Theorem 7.

(1) $SMON - TXT \subset MON^{\&} - TXT \subseteq MON^d - TXT \cap MON - TXT$

(2) $MON - TXT \# MON^d - TXT$

(3) $MON^d - TXT\#\ CONSERVATIVE\text{–}TXT$

Proof. Assertion (1) is proved via the following claims.

Claim 1. Let $\mathcal{L}$ be any indexed family and let M be any IIM inferring $\mathcal{L}$ strong–monotonically. Then M infers $\mathcal{L}$ dual monotonically.

Let $L \in \mathcal{L}$, let t be any text for L and let $(j_x)_{x \in N}$ be the sequence of M's hypotheses when successively fed with t. In accordance with the definition of $SMON - TXT$ we have $L(G_{j_x}) \subseteq L(G_{j_{x+r}}) \subseteq L$ for all $r \in N$. Consequently, $L(G_{j_x}) \cup L \supseteq L(G_{j_{x+r}}) \cup L$. But the latter inclusion is equivalent to $co - (L(G_{j_x}) \cup L) \subseteq co - (L(G_{j_{x+r}}) \cup L)$ which is equivalent to $co - L(G_{j_x}) \cap co - L \subseteq co - L(G_{j_{x+r}}) \cap co - L$. Hence, M works dual monotonically, too.

Therefore, $SMON - TXT \subseteq MON^{\&} - TXT$. The proper inclusion is established by the next claim.

Claim 2. $MON^{\&} - TXT \setminus SMON - TXT \neq \emptyset$

It suffices to show that the family $\mathcal{L}_{mon}$ introduced in the proof of Theorem 1 may be inferred monotonically as well as dual monotonically by a single IIM. As we have shown, the IIM M presented in the proof of Theorem 1 works monotonically. As an easy calculation shows, it works dual monotonically, too.

We proceed in showing assertion (2). Since we already know that $MON^d - TXT \setminus MON - TXT \neq \emptyset$ it remains to prove $MON - TXT \setminus MON^d - TXT \neq \emptyset$. However, we continue in demonstrating the slightly stronger result $MON - TXT \setminus MON^d - INF \neq \emptyset$. Let $\Sigma = \{a, b, c\}$. We set $L_1 = \{a\}^*$, and for $k > 1$ let $L_k = \{a^z \mid 1 \leq z \leq k\} \cup \{b^z \mid k < z\}$. Moreover, let $L_{k,m} = \{a^z \mid 1 \leq z \leq k\} \cup \{b^z \mid k < z \leq m\} \cup \{c^m\}$ for $m > k > 1$. Define $\mathcal{L}$ to be the indexed family of all these languages L_1, L_k and $L_{k,m}$.

Claim 3. $\mathcal{L} \in MON - TXT$

The wanted IIM M which monotonically identifies $\mathcal{L}$ on text works as follows: Let $L \in \mathcal{L}$, let t be any text for L, and let $x \in N$. As long as $t_x^+ \subseteq \{a\}^*$, the machine outputs a grammar for L_1. In case the first string s over $\{b\}$ does appear in the text and $t_x^+ \subseteq \{a\}^* \cup \{b\}^*$, the IIM computes $k = max\{z \mid a^z \in t_x^+\}$ and outputs a grammar for L_k. Finally, if a string c^m is presented, M calculates $k = max\{z \mid a^z \in t_x^+\}$ and outputs a grammar for $L_{k,m}$. It is straightforward to verify that $\mathcal{L} \in MON - TXT(M)$.

Claim 4. $\mathcal{L} \notin MON^d - INF$.

Suppose the converse, i.e., there is an IIM M which dual monotonically infers $\mathcal{L}$ on informant. Let i be any informant for $\{a\}^*$. Hence, there must be an x such that $j_x = M(i_x)$ and $L(G_{j_x}) = \{a\}^*$. Let $k = max\{|s| \mid s \in i_x^+ \cup i_x^-\}$. Then, we consider any informant $\tilde{i}$ for L_k that contains i_x as prefix. Since $L_k \in \mathcal{L}$, there has to be an y such that $j_y = M(\tilde{i}_y)$ and $L(G_{j_y}) = L_k$. Let $k = max\{|s| \mid s \in \tilde{i}_y^+ \cup \tilde{i}_y^-\}$. Obviously, $\tilde{i}_y$ is an initial segment of an informant i_{fool} for $L_{k,m}$. Then, M either does not work dual monotonically on i_{fool} or it fails to infer $L_{k,m}$. If M produces sometime the hypothesis j_x and thereafter j_y when processing i_{fool}, then $b^{m+1} \in co - L_1 \cap co - L_{k,m}$, but $b^{m+1} \notin co - L_k \cap co - L_{k,m}$ which violates the dual monotonicity requirement. Hence assertion (2) is proved.

Finally, we show (3). The part $MON^d - TXT \setminus CONSERVATIVE\text{–}TXT \neq \emptyset$ has been

proved above (cf. the proof of Theorem 4, assertion (2)). Conversely, $CONSERVATIVE\text{–}TXT \setminus MON^d - TXT \neq \emptyset$ is an immediate consequence of Claim 4 above. This proves the theorem.

q.e.d.

Notice the progression of the relationship of the dual constraints to the monotonicity constraints. For the strong–monotonic case, its dual version is more restrictive. Looking at the monotonic case, we saw that the two versions are incomparable. Finally, in the weak–monotonic case, the dual version is even more powerful.

Summarizing the results obtained above we may derive the following corollary.

Corollary 8.

(1) $FIN - TXT = SMON^{\&} - TXT \subset SMON - TXT \subset MON^{\&} - TXT$

(2) $WMON^{\&} - TXT = WMON - TXT \subset WMON^d - TXT$

Next we want to deal with the combination of monotonic and consistent inference. Considering a particular family of languages, the so–called pattern languages (cf. section 5) Jantke(1991B) has shown that weak–monotonicity and consistency may be combined without any limitations concerning the inference power, while strong–monotonicity and consistency cannot. However, the negative result is mainly based on the particular choice of the space of hypotheses, i.e., the set of all patterns, as a careful analysis of his proof shows. Nevertheless, we have been a bit surprised to obtain the following theorem:

Theorem 9.

(1) $WMON - CONS - TXT = WMON - TXT$

(2) $MON - CONS - TXT = MON - TXT$

(3) $SMON - CONS - TXT = SMON - TXT$

(4) $WMON^d - CONS - TXT = WMON^d - TXT$

(5) $MON^d - CONS - TXT = MON^d - TXT$

(6) $SMON^d - CONS - TXT = SMON^d - TXT$

Proof. We prove assertion (1). The rest can be obtained analogously. Obviously, $WMON - CONS - TXT \subseteq WMON - TXT$. For the opposite direction let $\mathcal{L}$ be any fixed indexed family of recursive languages and let M be an IIM such that $\mathcal{L} \subset WMON - TXT(M)$ with respect to some space $(G_j)_{j \in N}$ of hypotheses, i.e., $L \in WMON\text{–}TXT(M)$ for any $L \in \mathcal{L}$. We define an IIM $\hat{M}$ as follows:

Let t be any text for some language $L \in \mathcal{L}$ and let $x \in N$. When fed t_x the machine $\hat{M}$ simulates M's computation on t_y for all $y \leq x$. In case that M does not produce any hypothesis, $\hat{M}$ requests the next input and output nothing. Otherwise, let j_z be M's last guess. The machine $\hat{M}$ tests whether or not $t_x^+ \subseteq L(G_{j_z})$. In case it is, $\hat{M}$ outputs j_z and requests the next input. Otherwise, the IIM $\hat{M}$ outputs nothing and requests the next

input. By construction one directly obtains $L \in WMON-CONS-TXT(\hat{M})$. We omit the details.

q.e.d.

It remained open whether or not $WMON^d - TXT \subset LIM - TXT$. However, we obtained some results giving at least a partial answer to this question. Moreover, these results shed some more light on the problem what impact the choice of the space of hypotheses may have.

Our next theorem shows that dual weak–monotonic inference is eaxactly as powerful as learning in the limit provided the space of hypotheses may be arbitrarily chosen.

Theorem 10. *Let $\mathcal{L}$ be an indexed family such that $\mathcal{L} \in LIM - TXT$. Then there is a space $\mathcal{G} = (G_j)_{j \in N}$ of hypotheses such that $\mathcal{L} \in WMON^d - TXT$ with respect to $\mathcal{G}$.*

Proof. Let $\mathcal{L} \in LIM - TXT$. Then there are an IIM $\hat{M}$ as well as a space $\hat{\mathcal{G}} = (\hat{G}_j)_{j \in N}$ of hypotheses such that $\hat{M}$ infers $\mathcal{L}$ in the limit with respect to $\hat{\mathcal{G}}$. Without loss of generality we may assume that $\hat{M}$, after having output its first guess, does always output a hypothesis.

The wanted space $\mathcal{G}$ of hypotheses is a canonical enumeration of grammars generating all finite intersections of languages from $\mathcal{L}$. Consequently, $range(\mathcal{L}) \subseteq \{L(G_j) \mid j \in N\}$. We proceed in defining an IIM M inferring $\mathcal{L}$ dual weak–monotonically with respect to $\mathcal{G}$. Let $L \in \mathcal{L}$, let t be any text for L, and let $x \in N$.

$M(t_x) =$ "Simulate $\hat{M}$ on t_x. If $\hat{M}$ outputs its first guess j when successively fed with t_x, then output a canonical grammar for $L(\hat{G}_j)$. Otherwise, let k be M's last guess on t_{x-1}. Check whether or not $t_x^+ \subseteq L(G_k)$. In case it does not, output a canonical grammar for $\hat{M}(t_x)$.
Otherwise, continue as follows. Test whether or not $\hat{M}$ on t_x performs a mind change, say from j_{x-1} to j_x. In case it does not, output $M(t_{x-1})$.
Otherwise, output a canonical grammar for $L(G_k) \cap L(\hat{G}_{j_x})$."

It remains to prove that M dual weak–monotonically infers L.

Claim 1. M works dual weak–monotonically.

Suppose, M performs a mind change. In case this mind change is caused by inconsistency, we are done. By construction, if M performs a mind change, say from k to j, not forced by observing $t_x^+ \not\subseteq L(G_k)$, then j is a canonical grammar for some $L(G_k) \cap L(\hat{G}_{j_x})$, hence $co - L(G_k) \subseteq co - L(G_j)$.

Claim 2. If M converges on t, say to z, then $L = L(G_z)$.

Suppose the converse, i.e., $L \neq L(G_z)$. Then we immediately obtain $L \subset L(G_z)$, since otherwise M has to detect an inconsistency. Moreover, since M has converged, we may conclude that $\hat{M}$ has converged, too. Consequently, if $\hat{M}$ converges to k, then M either converges to a canonical grammar for $L(\hat{G}_k)$, or it converges to a canonical grammar for some $L(G_m) \cap L(\hat{G}_k)$, where m is M's last guess before it outputs z. In both cases we get that $\hat{M}$ fails to infer L, a contradiction.

Claim 3. M converges on t.

This is an immediate consequence of our construction. We omit the details.

q.e.d.

However, the space of hypotheses constructed above does not necessarily meet the requirement of class preserveness. Nevertheless, the same idea directly yields the following corollary.

Corollary 11. *If $\mathcal{L} \in LIM-TXT$, then there is an indexed family $\hat{\mathcal{L}}$ with $range(\mathcal{L}) \subset range(\hat{\mathcal{L}})$ such that $\hat{\mathcal{L}} \in WMON^d - TXT$.*

On the hand, in dealing with exact learning we have obtained the conjectured separation of dual weak–monotonic learning and identification in the limit.

Theorem 12. $ExactWMON - TXT \subset LIM - TXT$

Proof. Obviously, $ExactWMON^d - TXT \subseteq LIM - TXT$. Consequently, it suffices to prove that $LIM - TXT \setminus WMON^d - TXT \neq \emptyset$. In doing so, we mainly apply the proof technique used in Angluin (1980A) for showing $ExactCONSERVATIVE\text{-}TXT \subset LIM - TXT$. However, we have to strengthen it slightly in order to achieve the desired separation. Let $(M_j)_{j \in N}$ be Gold's (1967) standard enumeration of all inference machines over the alphabet $\Sigma = \{a, b\}$. Let c be Cantor's pairing function. We define an indexed family $\mathcal{L} = (L_{c(j,k)})_{j,k \in N}$ as follows. Let $j,\ k \in N$. We set $L_{c(j,1)} = \{a^j b^m \mid m \in N\}$, and for $k > 1$, $L_{c(j,k)}$ will be defined as a subset of $\{a^j b^m \mid m \in N\} \cup \{b^j a^m \mid m \in N\}$. This is done in accordance with the following construction.

Case 1: $a^j b^k \in L_{c(j,k-1)}$
Let t be the lexicographically ordered text for $L_{c(j,1)}$. Simulate M_j's computation on t for exactly k steps.

Subcase 1.1: Assume that there is an $x \leq k$ such that $M_j(t_x) = c(j,1)$ within at most k steps. Then, $a^j b^n \in L_{c(j,k)}$ if and only if $n \leq y$, where $y = \mu x[M_j(t_x) = c(j,1),\ x \leq k,\ within\ at\ most\ k\ steps]$. Furthermore, let us consider the text $\hat{t} = a^j b, a^j b^2 ..., a^j b^y, a^j b^y, ...$, i.e. $\hat{t}$ is obtained by extending t_y with its last string infinitely often. The following computation has to be performed in order to determine whether or not $b^j a^n \in L_{c(j,k)}$, for any $n \in N$. Simulate the computation of M_j on $\hat{t}_n$ for exactly n steps. If there is an z with $y \leq z \leq n$ such that $M_j(\hat{t}_z) = c(j,u)$ for some $u \geq k$ within at most n steps, then $b^j a^n \in L_{c(j,k)}$. Otherwise, $b^j a^n \notin L_{c(j,k)}$.

Subcase 1.2: There is no $x \leq k$ such that $M_j(t_x) = c(j,1)$ within at most k steps. Then, let $L_{c(j,1)} = L_{c(j,k)}$.

Case 2: $a^j b^k \notin L_{c(j,k-1)}$
In this case, set $L_{c(j,k)} = L_{c(j,k-1)}$.

Obviously, $\mathcal{L}$ is an indexed famlily. It remains to show that $\mathcal{L}$ may be used to separate $LIM - TXT$ from $ExactWMON^d - TXT$.

Claim A. $\mathcal{L} \notin ExactWMON^d - TXT$
Assume the opposite, i.e. there is an IIM M inferring $\mathcal{L}$ dual weak monotonically w.r.t. the space of hypotheses $\mathcal{L}$. Consequently, there has to be an index j such that $M = M_j$. If M_j on the lexicographically ordered text t for $L_{c(j,1)}$ never guesses $c(j,1)$, then it does fail to identify $L_{c(j,1)}$ from positive data. On the other hand, if M_j on the lexicographically ordered text t for $L_{c(j,1)}$ outputs $c(j,1)$ for the first time, say on t_y, then we have the

following situation. Let k be the number of steps it takes M_j to produce $c(j,1)$ on t_y. By construction, $L_{c(j,z)} = L_{c(j,1)}$ for all $z < k$ as well as $t_y^+ \subseteq L_{c(j,z)}$ for all $z \geq k$. Furthermore, if M_j never guesses $L_{c(j,z)}$ for any $z \geq k$ on the text $\hat{t} = a^j b \cdot a^j b^2, ..., a^j b^y, a^j b^y, ...$, then $L_{c(j,k)} = t_y^+$, and therefore the machine M fails to identify $L_{c(j,k)}$. Since $M_j(t_y) = c(j,1)$ as well as $L_{c(j,1)} \neq L_{c(j,k)}$, the IIM has to perform a mind change in order to infer $L_{c(j,k)}$ on the text $\hat{t}$. Due to our construction, M_j has to guess $c(j,z)$ for some $z \geq k$. However, this mind change cannot be caused by an inconsistency. Moreover, $L_{c(j,z)} \not\subseteq L_{c(j,1)}$. Therefore, M either does not identify $L_{c(j,k)}$ or otherwise, it does not work dual weak monotonically.

Claim B. $\mathcal{L} \in LIM - TXT$

We informally define an IIM M inferring $\mathcal{L}$ w.r.t. the space of hypotheses $\mathcal{L}$. Assume any $L \in \mathcal{L}$, any text t for L, and any $x \in N$. If x is sufficiently large, there has to be a j such that a string $a^j b^n$ occurs in t. M simulates M_j's computation on the text $\hat{t}_x = a^j b, a^j b^2, ...$ exactly x steps. In case M_j produces for the first time $c(j,1)$, say for a $y \leq x$ on $\hat{t}_y$ within $z \leq x$ steps, M checks whether or not $t_x^+ \subseteq L_{c(j,z)}$. In case it is, M outputs $c(j,z)$. Otherwise, M outputs $c(j,1)$. Finally, if M_j does not output $c(j,1)$ when fed with $\hat{t}_x$ in at most x steps, then M outputs $c(j,1)$. It is easy to see that M identifies $\mathcal{L}$ in the limit.

q.e.d.

The following figure summarizes the results obtained above. The lines between the identification types indicate inclusion, i.e., the lower type is properly contained in the upper one. Missing lines indicate incomparability of the collections of sets.

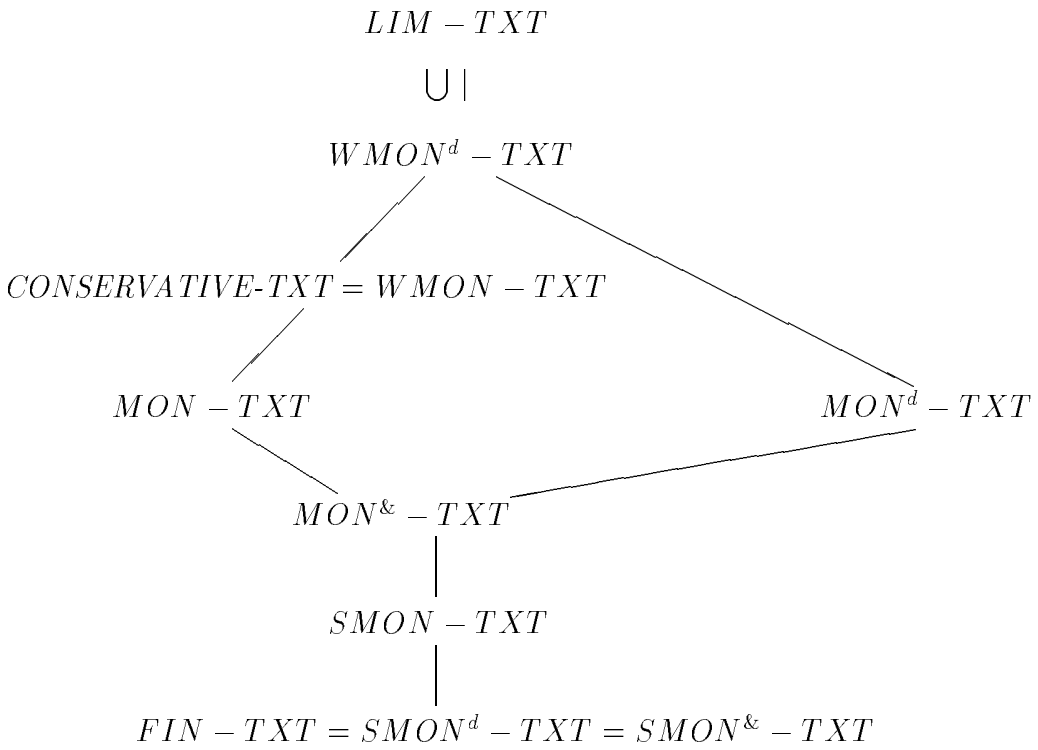

Figure 1

Finally, a closer look to the proof above shows that we mainly exploited the recursiveness of all languages in $\mathcal{L}$ as well as the fact that the IIMs are not required to produce on any input a hypothesis, i.e., our IIMs are not necessarily *responsive* (cf. Angluin (1980A)).

Nevertheless, if one weakens the condition $\mathcal{L} = \{L(G_j) \mid j \in N\}$ to $\mathcal{L} \subseteq \{L(G_j) \mid j \in N\}$, then consistency and responsiveness may be combined with all types of monotonic inference as well as with identification in the limit ($LIM - TXT$) without any limitation concerning the inference power. On the other hand, if one insists on spaces of hypotheses containing exactly those languages the indexed family to be inferred comprises, then IIMs that work consistently, responsively and monotonically are slightly weaker than monotonically working ones. The same is true for $LIM - TXT$–identification. This result impressively shows that the choice of the space of hypotheses may not only influence the inferring power at all but may or may not allow the combination of postulates of naturalness.

4. Monotonic Inference on Informant

The first theorem in the section again establishes a hierarchy between the different types of monotonicity.

Theorem 13.

(1) $SMON - INF \subset MON - INF \subset WMON - INF$

(2) $SMON^d - INF \subset MON^d - INF \subset WMON^d - INF$

(3) $MON - INF \# MON^d - INF$

(4) $SMON - INF \# SMON^d - INF$

Proof. $SMON{-}INF \subseteq MON{-}INF \subseteq WMON{-}INF$ is an immediate consequence of Definition 4. Moreover, in proving Theorem 1 we have shown that $MON - TXT \setminus SMON - INF \neq \emptyset$, hence we may conclude $SMON - INF \subset MON - INF$, since $MON - TXT \subseteq MON - INF$. For the demonstration of $WMON - INF \setminus MON - INF \neq \emptyset$ the reader is referred to the proof of Theorem 17, Claim C and D, where we even prove $WMON - TXT \setminus MON - INF \neq \emptyset$. Finally, assertion (3) and (4) have been proved in Lange and Zeugmann (1992B).

q.e.d.

Moreover, while the proper set inclusions remain valid if one considers only families of regular languages, the next theorem shows the power of weak–monotonic IIMs. However, the power of weak–monotonic IIMs again may be characterized as to coincide with the inferring capabilities of conservative ones.

Theorem 14.

(1) $FIN - INF \subset SMON - INF$

(2) $FIN - INF \subset SMON^d - INF$

(3) $WMON - INF = WMON^d - INF = WMON^{\&} - INF$

(4) $WMON^{\&} - INF = CONSERVATIVE{-}INF = LIM - INF$

Proof. The part $FIN - INF \subseteq SMON - INF$ is again obvious. The propper set inclusion is obtained by observing that $\mathcal{L}_{finvar} \notin FIN - INF$, where $\mathcal{L}_{finvar} = \{L \cup L_{fin} \mid L^M_{fin} \in \mathcal{L}_{fin}\}$, i.e., the family defined before presenting Definition 5. On the other hand, $\mathcal{L}_{finvar} \in SMON - TXT$ as we have shown above. Hence assertion (1) follows. The second assertion is an immediate consequence of Theorem 13, part (4). The proof of (3) and (4) can be found in Lange and Zeugmann (1992B).

q.e.d.

Summarizing the results above one easily derives the following corollary.

Corollary 15.

(1) $SMON - INF \subset MON^{\&} - INF$

(2) $SMON^d - INF \subset MON^{\&} - INF$

(3) $FIN - INF = SMON^{\&} - INF$

The following figure gives a first overview concerning monotonic and dual monotonic inference on informant.

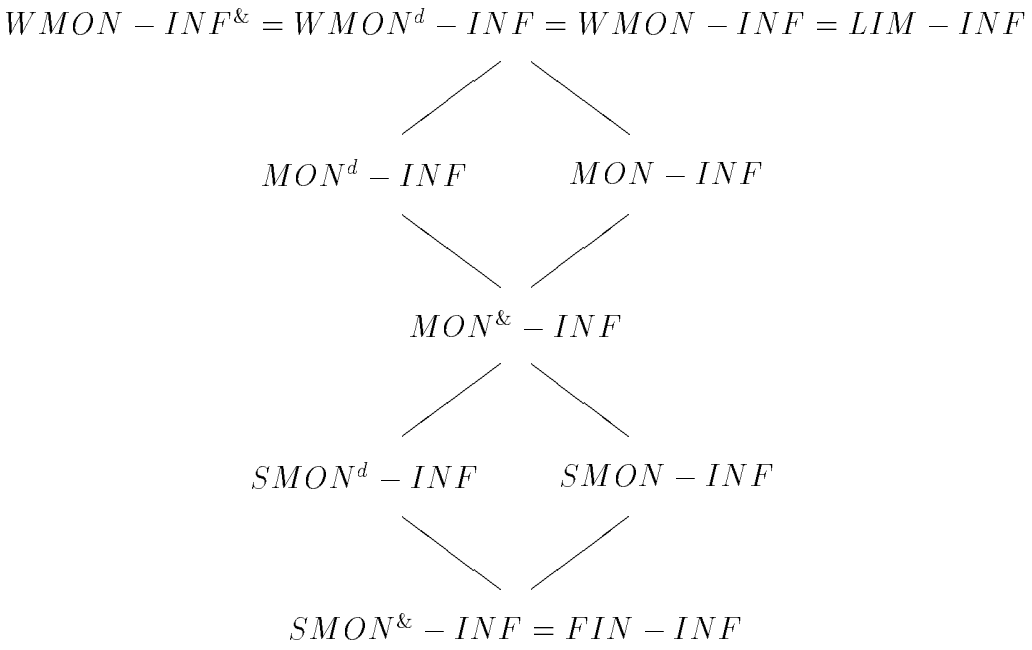

Figure 2

All lines between identification types indicate inclusions in the sense that the upper type always properly contains the lower one. If two identification types are not connected by ascending lines, then they are incomparable.

The next theorem shows that we again may combine monotonicity and consistency without limiting the learning power.

Theorem 16.

(1) $SMON - CONS - INF = SMON - INF$

(2) $MON - CONS - INF = MON - INF$

(3) $WMON - CONS - INF = WMON - INF$

(4) $SMON^d - CONS - INF = SMON^d - INF$

(5) $MON^d - CONS - INF = MON^d - INF$

(6) $WMON^d - CONS - INF = WMON^d - INF$

Proof. This theorem can be shown analogously as Theorem 9.

q.e.d.

Moreover, in our setting, i.e., considering only indexed families of recursive languages, we even obtain that $LIM - INF = CONS - INF$.

Now we want to compare monotonic and dual monotonic inference from positive data with monotonic and dual monotonic inference from both, positive and negative data. In his seminal paper Gold (1967) has shown that the inference from positive data alone is much weaker than inference on informant. Consequently, one might well expect that monotonic inference on text is less powerful than learning monotonically on informant. However, the more interesting question is whether one can strengthen the monotonicity requirement in case one changes from positive data to positive and negative data. The opposite direction of this problem is whether the weakening of the notion of monotonicity allows the inference of sets of languages on text which may only be inferred on informant in the stronger monotonic case. Our next theorem shows that all these things are almost always different ones, i.e., in general one cannot trade monotonicity versus information presentation.

Theorem 17.

(1) $MON - INF \# WMON - TXT$

(2) $MON - INF \# LIM - TXT$

(3) $SMON - INF \# MON - TXT$

(4) $FIN - INF \# SMON - TXT$

Proof. The first part we prove here is $MON - INF \setminus LIM - TXT \neq \emptyset$. We set $L_1 = \{a\}^*$, $L_k = L_1 \setminus \{a^k\}$ for $k > 1$, and define $\mathcal{L} = L_1, L_2, \ldots$. Obviously, $\mathcal{L}$ is an indexed family.

Claim A. $\mathcal{L} \in MON - INF$

The wanted IIM M works as follows: Initially, it outputs a canonical grammar j_1 for L_1. As long as some $(a^k, -)$ does not appear in the informant, the machine M outputs j_1. In case it does, M performs a mind change and outputs a canonical grammar for L_k. This

hypothesis is then repeated in any subsequent step. Since $L_1 \cap L_k = L_k$ for all $k \geq 1$, the machine M works monotonically.

Claim B. $\mathcal{L} \notin LIM - TXT$
Due to Theorem 1 of Angluin(1980A) it suffices to show that there is no finite tell–tale for L_1. Suppose the converse, i.e., there is a recursively enumerable finite set T_1 satisfying:

(1) $T_1 \subset L_1$, and

(2) for all $j > 1$, if $T_1 \subseteq L_j$, then $L_j \not\subset L_1$.

Let $z = max\{|s| \mid s \in T_1\}$. Then, in accordance with the definition of the languages L_k we immediately get $L_{z+1} \supseteq T_1$. Moreover, $L_{z+1} \subset L_1$, yielding a contradiction to condition (2). This proves $MON - INF \setminus LIM - TXT \neq \emptyset$.

Next we show $WMON - TXT \setminus MON - INF \neq \emptyset$ in using the indexed family $\mathcal{L}_{wmon}$ mentioned in the proof of Theorem 1.

Claim C. $\mathcal{L}_{wmon} \in WMON - TXT$
The wanted IIM M is informally defined as follows: Let $L \in \mathcal{L}_{wmon}$, and let t be any text for L. M initially outputs a canonical grammar for the languaguge $\{a\}^*$. In case the first string b^z appears in t the machine M outputs a grammar for L_z. This guess remains unchanged until

(α) there appears a sting b^k in the text t with $k < z$ or

(β) there appears a string a^m in the text t with $z < m$.

If (α) happens first, then M outputs a grammar for L_k, in case (β) happens first $M's$ new output is $L_{z,m}$, and finally, if (α) happens after (β) has already happened, then M produces a canonical grammar for $L_{k,m}$.
Looking at the definition of M we immediately observe that M changes its mind only in case if its current guess is inconsistent with the new data. Consequently, $\mathcal{L}_{wmon} \in WMON - TXT(M)$.
Finally we have to prove the following claim:

Claim D. $\mathcal{L}_{wmon} \notin MON - INF$
Suppose the converse, i.e., assume $\mathcal{L}_{wmon} \in MON - INF(M)$ for some IIM M. Let i be any informant for $\{a\}^*$. Since $L_0 \in \mathcal{L}_{wmon}$ there must be an x such that $j_x = M(i_x)$ and $L(G_{j_x}) = L_0$. Next we successively enlarge i_x by $(b^z, +)$, where $z \geq y = max\{|s| \mid s \in i_x^+ \cup i_x^-\} + 1$. Consequently, all i_{x+k} are initial segments of an informant for L_y. Hence there must be a number k such that M on i_{x+k} outputs a grammar j_{x+k} being correct for L_y. But now we may enlarge i_{x+k} in a canonical manner to an informant i_{fool} for $L_{y,m}$ where $m = max\{|s| \mid s \in i_{x+k}^+ \cup i_{x+k}^-\} + 1$. It is easy to see that M either does not work monotonically on i_{fool} or it does not infer $L_{y,m}$. This proves the claim, and so we have proved assertion (1) and (2).

Next we prove (3) and (4). The part $MON - TXT \setminus SMON - INF \neq \emptyset$ has been shown in the proof of Theorem 1. In order to complete the proof it suffices to show that $FIN - INF \setminus MON - TXT \neq \emptyset$. This is done via the family $\mathcal{L}_{dmon}$ defined as follows.

Let $k \in N$, and set $L_k = \{a\}^* \setminus \{a^k\}$. We define $\mathcal{L}_{dmon}$ to be the family of all languages L_k. Obviously, $\mathcal{L}_{dmon}$ is an indexed family. First we show that $\mathcal{L}_{dmon} \notin MON - TXT$. Suppose it is, i.e., there is an IIM M that identifies $\mathcal{L}_{dmon}$ on text. Let us consider M's behavior on the following text t_{fool}. We feed $a^2, a^3, ...$ to M until it outputs a guess j_{z-1} (say on $a^2, a^3, ..., a^z$) such that $L(G_{j_{z-1}}) = L_1$. In case it does not, we are done, since M does not identify L_1 on it lexicographically ordered text. Then we input $a, a^{z+2}, a^{z+3}, ...$ until M produces a correct guess j_r for L_{z+1}. Again, there has to be an $r \geq z+2$ such that M on $a^2, a^3, ..., a^z, a, a^{z+2}, a^{z+3}, ..., a^r$ outputs j_r, since otherwise M never infers L_{z+1}. But now we may proceed as follows. We set $t_{fool} = a^2, a^3, ..., a^z, a, a^{z+2}, ..., a^r, a^{r+2}, a^{r+3}, ...$, i.e., t_{fool} is a text for L_{r+1}. Since $a^{z+1} \in L_1 \cap L_{r+1}$ but $a^{z+1} \notin L_{z+1} \cap L_{r+1}$, we get $L_1 \cap L_{r+1} \not\subseteq L_{z+1} \cap L_{r+1}$. Consequently, M does not work monotonically on t, a contradiction.

It remains to show that $\mathcal{L}_{dmon} \in FIN - INF$. The IIM M finitely inferring $\mathcal{L}_{dmon}$ on informant works as follows: As long as i_x does not contain a pair $(a^k, -)$, the machine M outputs nothing. In case a pair $(a^k, -)$ appears, M outputs a canonical grammar for L_k and stops.

q.e.d.

Theorem 17 has the following direct consequences which are summarized in the next corollary.

Corollary 18.

(1) $SMON - TXT \subset SMON - INF$

(2) $MON - TXT \subset MON - INF$

(3) $WMON - TXT \subset WMON - INF$

(4) $LIM - TXT \subset LIM - INF$

(5) $FIN - TXT \subset FIN - INF$

Next we ask whether assertion (2) of the latter theorem can be sharpened to $SMON - INF \setminus LIM - TXT \neq \emptyset$. The answer to this question put the hardness of strong–monotonic inference in the right perspective.

Theorem 19. $SMON - INF \subset WMON - TXT$

Proof. It suffices to show $SMON - INF \subseteq WMON - TXT$ since by Theorem 17, assertion (1), one directly obtains $WMON - TXT \setminus SMON - INF \neq \emptyset$. Let $\mathcal{L} \in SMON - INF(M)$, for some IIM M. Without loss of generality we may assume M to be total, i.e., to be defined on any finite sequence of strings from $\Sigma^* \times \{+, -\}$. This can be seen as follows.

Let $\hat{M}$ be any IIM inferring $\mathcal{L}$ strong-monotonically on informant. Moreover, let $i_x = (s_1, b_1), ..., (s_x, b_x)$ be any finite sequence. Then the IIM M first checks whether or not there is a language $L_k \in \mathcal{L}$ such that $k \leq x$, $i_x^+ \subseteq L_k$, and $i_x^- \subseteq co - L_k$. In case such L_k does not exists, M requests the next input. Otherwise, M on input i_x behaves exactly as $\hat{M}$ does. Note that in the latter case $\hat{M}$ has to be defined, since otherwise it would fail on a possible initial segment of some informant for L_k. Moreover, M infers $\mathcal{L}$ strong-monotonically. Let $L \in \mathcal{L}$, $y = \mu z[L_z = L]$, and i any informant of L. Consequently, on

input i_x, for any $x \geq y$, M finds at least one language L_k such that $k \leq x$, $i_x^+ \subseteq L_k$, and $i_x^- \subseteq co - L_k$, i.e., it works exactly as $\hat{M}$ does. Furthermore, on input i_x, for any $x < y$, it either behaves as $\hat{M}$ does or it possibly ships over some hypotheses $\hat{M}$ possibly might have computed. Since set inclusion is a transitive relation, M works strong monotonically.

Let us continue. We have to construct an IIM M' identifying $\mathcal{L}$ weak-monotonically on text. This is done as follows: Let $L \in \mathcal{L}$ and let $t = (s_j)_{j \in N}$ be an arbitrarily fixed text for L. Then the wanted machine M' is defined as follows:

$M'(t_x) =$ "Rearrange the strings contained in t_x in lexicographical order without repititions. Let $t'_k = w_1, ..., w_k$, $k \leq x$ the sequence obtained.
Compute $A_x := \{(w_1, +), ..., (w_k, +)\} \cup \{(v, -) \mid v \in \Sigma^* \setminus t_x^+, \quad |v| \leq max\{x, |w_k|\}\}$
Determine i_z by rearranging the elements of A_x in lexicographical order with respect to the first component.
(* Note that i_z is not necessarily an initial segment of an informant of L, since it may contain some pair $(v, -)$ such that $v \in L$. However, if x is large enough, then there is an $l \in N$ such that i_m is an initial segment of an informant for L for any $m \leq l$ *)
Compute $M(i_0)$, $M(i_1)$, ... until M outputs the first hypothesis j_r, where $r \leq z$, satisfying $i_z^+ \subseteq L(G_{j_r})$ and $i_r^- \subseteq co - L(G_{j_r})$.
(*Observe that every string $(w, +) \in i_z^+$ belongs indeed to L by our construction*)
In case j_r has been found, output it. Otherwise, output nothing, and request t_{x+1}."

It remains to show that $L \in WMON - TXT(M')$. We have to prove that

(1) M' converges to a number j such that $L = L(G_j)$, and

(2) M' works weak–monotonically.

In order to show (1) remember that M works strong–monotonically on every informant for L. Consequently, M works strong–monotonically on the lexicographically ordered informant i. Hence, there is a minimal index m such that $M(i_m) = j_m$ and $L(G_{j_m}) = L$. Moreover, since t is a text for L there is an index x satisfying $i_m^+ \subseteq t_{x+k}^+$ and $i_m^- \subseteq range(i_{z+k})$ for all $k \geq 1$, where i_{z+k} is the initial segment of an informant obtained from t_{x+k} as described above. Since $L(G_{j_m}) = L$ we get:

$$i_{z+k}^+ \subseteq L(G_{j_m}) \; and \; i_m^- \subseteq co - L(G_{j_m}) \; for \; every \; k \geq 1.$$

Thus, $M'(t_x) = M'(t_{x+k})$ for all $k \geq 1$, i.e., M' converges.

We proceed in showing (2). Since M works strong–monotonically on i_m we have: Whenever M on i_l, $l \leq m$, outputs a guess j_l, then $L(G_{j_l}) \subseteq L$.
Now it suffices to prove that every mind change of M' is forced by an inconsistency with the text t. Let j_r be guess produced by M', i.e., in simulating M an initial segment i_r has been found such that $M(i_r) = j_r$ and $i_z^+ \subseteq L(G_{j_r})$ as well as $i_r^- \subseteq co - L(G_{j_r})$. We distinguish the following two cases:
Case (α): i_r is an initial segment of i_m

Then $L(G_{j_r}) \subseteq L$. Consequently, if j_r is not the hypothesis M' converges to, then there has to be a k such that $i_{z+k}^+ \not\subseteq L(G_{j_r})$. But that means the text t has to contain

some word $w \notin L(G_{j_r})$. Hence, this mind change is forced by an inconsistency.
Case (β): i_r is not an initial segment of i_m

Then i_r must contain a string $(v,-)$ such that $v \in L$. Since $i_r^- \subseteq co-L(G_{j_r})$ it follows that $v \notin L(G_{j_r})$. Consequently, in this case j_r is changed, if either a string s appears in t such that $(s,-) \in i_r^-$ or $s \in i_z^+$ and $s \notin L(G_{j_r})$. In both cases the mind change is due to inconsistency.

q.e.d.

Considering dual strong–monotonic inference on informant the situation is totally different.

Theorem 20. $SMON^d - INF \setminus LIM - TXT \neq \emptyset$

Proof. This proof is done via the indexed family $\mathcal{L}$ introduced in the proof of Theorem 17. Since we already know that $\mathcal{L} \notin LIM - TXT$, it remains to show that $\mathcal{L}$ can be inferred dual strong–monotonically on informant. The desired IIM M witnessing $\mathcal{L} \in SMON^d - INF$ is defined as follows: Let $L \in \mathcal{L}$ and let i be any informant for L. Initially, M outputs a canonical grammar j_1 for L_1. This guess is repeated until a pair $(a^k,-)$ does appear in i. Then M chnges its mind to a canonical grammar j_k for L_k and outputs subsequently j_k.

Obviously, M infers $\mathcal{L}$. Since $co-L_1 \subset co-L_k$ for any $k>1$, M works dual strong–monotonically, too.

q.e.d.

Theorem 20 directly yields the following consequences.

Corollary 21.

(1) $SMON^d - TXT \subset SMON^d - INF$

(2) $MON^d - TXT \subset MON^d - INF$

(3) $WMON^d - TXT \subset WMON^d - INF$

Finally, summarizing the results previously obtained we may derive the following corollary.

Corollary 22.

(1) $SMON^d - TXT \subset FIN - INF$

(2) $SMON^d - INF \# MON^d - TXT$

(3) $MON^d - INF \# LIM - TXT$

(4) $MON^d - INF \# WMON^d - TXT$

Proof. The only part not yet proved is $MON^d - TXT \setminus SMON^d - INF \neq \emptyset$. We define an indexed family over $\{a,b\}$ as follows: We set $L_1 = \{a\}^*$, $L_k = \{a^n \mid 1 \leq n \leq k\} \cup \{b^{k+1}\}$ with $k \geq 2$, and $L_{k,m} = L_k \cup \{a^m\}$ with $m > k \geq 2$. Assume $\mathcal{L}$ to be any appropriate enumeration of all these languages. An IIM M which dual monotonically infers $\mathcal{L}$ may

work as follows: Let $L \in \mathcal{L}$ and let t be any text for L. As long as $t_x^+ \subseteq \{a\}^*$ the machine outputs a grammar for L_1. M performs a mind change when both, b^{k+1} and a^k have been appeared in the text. Then M outputs a grammar for L_k. Subsequently, the IIM M test whether or not a string a^m with $m > k$ is contained in the textsegment it has been fed with. In case it is, it changes its mind to $L_{k,m}$. This hypothesis is then repeated in any subsequent step. Otherwise, it repeats its last guess. Obviously, M identifies $\mathcal{L}$. It remains to show that M works dual monotonically as well. By construction, M performs at most two mind changes, i.e., it eventually outputs j_x for L_1, j_{x+z} for L_k and finally j_{x+z+y} for $L_{k,m}$. Then we have $co-L_1 \cap co-L_{k,m} \subseteq co-L_k \cap co-L_{k,m} \subseteq co-L_{k,m}$, since $co-L_{k,m} = co-L_k \setminus \{a^m\}$. Hence $co-L_k \cap co-L_{k,m} = co-L_{k,m}$, i.e, M does work dual monotonically.

In Lange and Zeugmann (1992B) we have shown that $\mathcal{L} \notin MON-INF$. This completes the proof.

q.e.d.

Summarizing these results with those ones obtained previously, we get the following figures.

$$FIN-TXT \subset SMON-TXT \subset MON-TXT \subset WMON-TXT \subset LIM-TXT$$

$$\cap \qquad\quad \cap \qquad\quad \cap \qquad\quad \cap \qquad\quad \cap$$

$$FIN-INF \subset SMON-INF \subset MON-INF \subset WMON-INF = LIM-INF$$

Figure 3

The next figure shows the similarities as well as the differences between the types of monotonic and of dual monotonic inference.

$$FIN-TXT = SMON^d-TXT \subset MON^d-TXT \subset WMON^d-TXT \subseteq LIM-TXT$$

$$\cap \qquad\quad \cap \qquad\quad \cap \qquad\quad \cap \qquad\quad \cap$$

$$FIN-INF \subset SMON^d-INF \subset MON^d-INF \subset WMON^d-INF = LIM-INF$$

Figure 4

After having clarified the principal relations between the different notions of monotonicity we want to deal with the question what "natural" sets of languages may or may not be inferred monotonically. This is done in the next section.

5. Monotonic Learning of Natural Families of Languages

In this section we mainly deal with the question whether sets of pattern languages may be learnt strong–monotonically or monotonically on text or on informant. Pattern languages have been introduced by Angluin (1980B), thereby proving that the whole class of pattern languages can be inferred in the limit from positive data. Subsequently, Shinohara (1982) dealt with polynomial time learnability of subclasses of pattern languages. Nix (1983) outlined interesting applications of pattern inference algorithms. Recently, Kearns and Pitt (1989) as well as Ko et al. (1990) studied intensively pattern inference from positive and negative data in the PAC–learning model.

So let us define what are a pattern and a pattern language. Let Σ be any non–empty finite alphabet, and let $X = \{x_1, x_2, ...\}$ a set of variables. *Patterns* are non–empty strings of variables and constants from Σ, e.g. x_1x_2, $ax_1x_7bx_3$ are patterns. $L(p)$, the language generated by pattern p is the set of strings which can be obtained by substituting non–null strings from Σ^* for the variables of the pattern p. PAT denotes the set of all pattern languages. Finally, let DPAT be the family of all languages L for which there are patterns p_{i_1}, p_{i_2}, ..., p_{i_k} such that $L = \bigcap_{j=1}^{n} L(p_{i_j})$.

Using the techniques developed in Lange and Wiehagen (1990) one gets the following theorem:

Theorem 23.

(1) $PAT \in FIN - INF$

(2) $PAT \notin FIN - TXT$

Proof. In order to prove assertion (1) let $L \in PAT$ and let i be any informant for L. The wanted IIM M is informally defined as follows:

> M requests more and more inputs i_x until it finds the $k \in N$ such that the following conditions are satisfied:
>
> (α) Every $s \in \Sigma^*$ with $|s| < k$ belongs to i_x^-.
>
> (β) There is a $s \in \Sigma^*$ of length k that is contained in i_x^+, and all $s \in \Sigma^*$ of length k are classified in i_x.
>
> Having found this k the machine M starts the procedure described in Lange and Wiehagen (1990) with all strings $s \in L$ of length k and generates the uniquely determined canonical pattern p such that $L = L(p)$. Finally, it outputs p and stops.

The proof of assertion (2) can be directly obtained by observing that every text for any pattern language L is also a text for $L(x_1)$. Hence, $L(x_1)$ cannot be finitely inferred on text. We omit the details.

q.e.d.

On the other hand, Jantke (1991B) pointed out that $PAT \notin SMON-CONS-TXT$. However, this result is mainly based on the requirement that in his setting any IIM is only allowed to output patterns. If the space of hypotheses is restricted to PAT, then one can reduce the question whether at least $PAT \in SMON - TXT$ to the decidability of inclusion of pattern languages, i.e., to decide whether or not $L(p) \subseteq L(q)$ for any given patterns p and q. Unfortunately, the latter problem, first stated in Angluin (1980B), is open.

Therefore it is only natural to ask whether an appropriate choice of the space of hypotheses may help to overcome the difficulties in designing an IIM that infers PAT strong–monotonically. Note that the next theorem is mainly based on an idea communicated to us by H.R. Beick (1991).

Theorem 24. *There is a space of hypotheses such that* $DPAT \in SMON - TXT$.

Proof. The space of hypotheses is a subset of all context–sensitive grammars obtained as follows: It is not hard to see that there is an effective procedure translating every pattern p into a context–sensitive grammar G_p such that $L(p) = L(G_p)$. Moreover, the set of all context–sensitive grammars is closed under intersection (cf. [6]), i.e., given two context–sensitive grammars G_1 and G_2 one can effectively construct a grammar $G_{1,2}$ such that $L(G_{1,2}) = L(G_1) \cap L(G_2)$. Consequently, one can effectively construct a recursively enumerable family $(G_j)_{j \in N}$ of grammars such that $DPAT = \{L(G_j) \mid j \in N\}$.

We define an IIM M inferring $DPAT$ strong–monotonically on text as follows: Let $L \in DPAT$ and let t be any text for L. For any $x \in N$ we set:

$M(t_x) =$ "Compute $z = min\{|s| \mid s \in t_x^+\}$ and generate all canonical patterns $p_1, ..., p_k$ with $|p_i| \leq z$.
Test for all these patterns if $t_x^+ \subseteq L(p_i)$ for $i = 1, ...k$ and let I_x be the set of indices fulfilling the test.
Output a canonical grammar j_x for $\bigcap_{i \in I_x} L(p_i)$."

It remains to show that $DPAT \in SMON - TXT(M)$. Let $L \in DPAT$, i.e., there are patterns $q_1, ..., q_l$ such that $L = \bigcap_{j=1}^{l} L(q_j)$. Let $m = min\{|s| \mid s \in L\}$. Consequently, $|q_j| \leq m$ for all $j = 1, ...l$. Moreover, since t is a text for L we directly obtain that $m \leq z$ for all initial segments t_x of t and that $t_x^+ \subseteq L(q_j)$ for all $j = 1, ...l$. Thus $\{1, ..., l\} \subseteq I_x$ for every $x \in N$. Hence, every $j_x = M(t_x)$ satisfies $L(G_{j_x}) \subseteq L$. Furthermore, since $I_{x+k} \subseteq I_x$ for every $k \in N$ we additionally get $L(G_{j_x}) \subseteq L(G_{j_{x+k}}) \subseteq L$ for all $k \in N$. Consequently, M works strong–monotonically. Finally, M obviously converges, since $I_0 \setminus \{1, ...l\}$ is finite. Let j_{fin} be the limit of the sequence of hypotheses. Suppose $L(G_{fin}) \subset L$. Then there is at least a string $s \in L \setminus L(G_{fin})$. On the other hand, there must be an $x \in N$ such that $s \in t_x^+$, since t is a text for L. Due to the construction it must hold that $t_x^+ \subseteq L(p_i)$ for every $i \in I_x$, in particular $s \in L(p_i)$ for all $i \in I_x$. Hence $s \in L(G_{fin})$. Consequently, M converges to a correct grammar.

q.e.d.

On the other hand, at first glance it might seem that the decidability of $L(G) \subseteq L(G')$ may be generally helpful in proving that families of languages are monotonically learnable. However, decidability of inclusion does *not* guarantee monotonic learnability, even *on informant*.

Theorem 25. *The set of all regular languages cannot be monotonically inferred on informant.*

Proof. The proof can be done via the languages of the indexed family $\mathcal{L}_{wmon}$ introduced in the demonstration of Theorem 1. All these languages are regular. However, in an analogous manner to the proof of Claim D in the demonstration of Theorem 8 one can show that there is no IIM M inferring all the regular languages monotonically on informant. We omit the details.

q.e.d.

Let us finish this section with the remark that several problems remained open. We shall discuss them in the next section.

6. Conclusions and Open Problems

We have studied the power of strong–monotonic, monotonic and weak–monotonic as well as of dual strong–monotonic, dual monotonic and dual weak–monotonic IIMs in the setting of algorithmic language learning. All these notions have been related one to the other and several new hierarchies could been proven. Although all these notions seem to be quite natural ones, in general, dual strong–monotonicity, strong–monotonicity as well as monotonicity and dual monotonicity lead to a severe restriction of the inference power. This actually shows that learning processes have to be performed, at least to some extent, non–monotonically or at least weak–monotonincally or dual weak–monotonincally. Moreover, as our results show, if one deals with any type of monotonically or dual monotonically working IIMs the choice of the space of hypotheses is of great influence to what actually can be inferred. Generally, when dealing with types of monotonic or dual monotonic learning *exactness* does considerably restrict the inference power. Furthermore, some of our sepapration results are heavily dependend on *class preserving* spaces of hypotheses, e.g., $MON-TXT \subseteq WMON-TXT$ and $SMON^d-TXT \subseteq FIN-TXT$. Consequently, further work is necessary to study systematically the impact of different requirements to the admissible spaces of hypotheses.

Moreover, it seems to be challenging to combine monotonic language learning with other postulates of naturalness, e.g., with iterative inference introduced by Wiehagen (1976). Some results along this line may be found in Lange and Zeugmann (1992C).

Looking at the results presented above the most intriguing open question is whether or not $WMON^d-TXT \subset LIM-TXT$. It remained also open if $MON^{\&}-TXT \subset MON^d-TXT \cap MON-TXT$ as well as $SMON^{\&}-INF \subset SMON^d-INF \cap SMON-INF$ and $MON^{\&}-INF \subset MON^d-INF \cap MON-INF$. In the context of inference of natural families of languages the most interesting open question is whether $PAT \in SMON-TXT$, if the space of hypotheses is PAT.

Acknowledgement

The first two authors heartily thank Rolf Wiehagen for many inspiring discussions, and especially, for pointing out an error in an earlier version of this paper.

7. References

[1] D. Angluin, Inductive Inference of Formal Languagues from Positive Data, *Information and Control* **45** (1980A) 117 - 135.

[2] D. Angluin, Finding Patterns Common to a Set of Strings, *Journal of Computer and System Sciences* **21** (1980B) 46 - 62.

[3] D. Angluin and C.H. Smith, Formal Inductive Inference, St.C. Shapiro, ed., *Encyclopedia of Artificial Intelligence, Vol. 1* (Wiley-Interscience Publication, New York, 1987) 409 - 418.

[4] Ya.M. Barzdin, Inductive Inference of Automata, Functions and Programs, *Proc. Int. Congress of Math., Vancouver* (1974) 455 - 460.

[5] H.R. Beick (1991), Personal Communication.

[6] W. Bucher and H. Maurer, *Theoretische Grundlagen der Programmiersprachen, Automaten und Sprachen* (Bibliographisches Institut AG, Wissenschaftsverlag, Zürich, 1984).

[7] J. Case, The Power of Vacillation, *Proc. 1st Workshop on Computational Learning Theory*, D. Haussler and L. Pitt, eds., (Morgan Kaufmann Publishers Inc., 1988) 196 -205.

[8] J. Case and C. Lynes, Machine Inductive Inference and Language Identification, *Proc. Automata, Languages and Programming, Ninth Colloquim, Aarhus, Denmark*, M.Nielsen and E.M. Schmidt, eds., Lecture Notes in Computer Science Vol. 140 (Springer, Berlin, 1982) 107 -115.

[9] M. Fulk, Prudence and other Restrictions in Formal Language Learning, *Information and Computation* **85** (1990) 1 - 11.

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

[11] M. Kearns and L. Pitt, A Polynomial-time Algorithm for Learning k-variable Pattern Languages from Examples, *Proc. 2nd Workshop on Computational Learning Theory*, R . Rivest, D. Haussler and M.K. Warmuth, eds.,(Morgan Kaufmann Publishers Inc., 1989) 57 - 70.

[12] S. Jain and A. Sharma, Language Learning by a "Team", *Proc. Automata, Languages and Programming, 17th International Colloquium, Warwick University, England*, M.S. Paterson ed., Lecture Notes in Computer Science Vol. 443 (Springer, Berlin, 1990) 153 - 166.

[13] K.P. Jantke, Monotonic and Non–monotonic Inductive Inference, *New Generation Computing* **8** (1991A) 349 - 360.

[14] K.P. Jantke, Monotonic and Non–monotonic Inductive Inference of Functions and Patterns, *Proc. First International Workshop on Nonmonotonic and Inductive Logics, December 1990, Karlsruhe*, J. Dix, K.P. Jantke and P.H. Schmitt eds., Lecture Notes in Artificial Intelligence Vol. 543 (Springer, Berlin, 1991B) 161 - 177.

[15] K. Ko, A. Marron and W.G. Tzeng, Learning String Patterns and Tree Patterns From Examples, *Proc. 7th Conference on Machine Learning* (1990) 384 - 391.

[16] S. Lange and R. Wiehagen, Polynomial–Time Inference of Arbitrary Pattern Languages, *New Generation Computing* **8** (1991) 361 - 370.

[17] S. Lange and T. Zeugmann, Learning Recursive Languages with Bounded Mind Changes, GOSLER–Report xx/92, FB Mathematik und Informatik, TH Leipzig, 1992A.

[18] S. Lange and T. Zeugmann, Monotonic Language Learning on Informant, *Journal of Experimental and Theoretical Artificial Intelligence* (1992B), to appaer.

[19] S. Lange and T. Zeugmann, Types of Monotonic Language Learning and Their Characterization, *Proc. 5th Annual ACM Workshop on Comutational Learning Theory, July 27 - 29, Pittsburgh* (ACM Press, 1992C) 377 - 390.

[19] R.P. Nix, Editing by Examples, Department of Computer Science, Yale University, Technical Report 280, 1983.

[20] D. Osherson, M. Stob and S. Weinstein, *Systems that Learn, An Introduction to Learning Theory for Cognitive and Computer Scientists* (MIT-Press, Cambridge, Massachusetts, 1986).

[21] T. Shinohara, Polynomial Time Inference of Extended Regular Pattern Languages, *Proc. RIMS Symposia on Software Science and Engineering, Kyoto*, Lecture Notes in Computer Science Vol. 147 (Springer, Berlin, 1982) 115 - 127.

[22] R. Solomonoff, A Formal Theory of Inductive Inference, *Information and Control* **7** (1964) 1 - 22, 234 - 254.

[23] B.A. Trakhtenbrot and Ya.M. Barzdin, *Konetschnyje Awtomaty (Powedenie i Sintez)*, (Nauka, Moskwa, 1970) (in Russian)
english translation:*Finite Automata–Behavior and Synthesis, Fundamental Studies in Computer Science 1* (North–Holland, Amsterdam, 1975).

[24] R. Wiehagen, Limes–Erkennung rekursiver Funktionen durch spezielle Strategien, *Journal Information Processing and Cybernetics (EIK)* **12** (1976) 93 - 99.

[25] R. Wiehagen, (1977), Identification of Formal Languages, *Proc. Mathematical Foundations of Computer Science, Tatranska Lomnica*, J. Gruska ed., Lecture Notes in Computer Science Vol. 53 (Springer, Berlin, 1977) 571 - 579.

[26] R. Wiehagen, (1991), A Thesis in Inductive Inference, *Proc. First International Workshop on Nonmonotonic and Inductive Logic, December 1990, Karlsruhe*, J.Dix, K.P. Jantke and P.H. Schmitt eds., Lecture Notes in Artificial Intelligence Vol. 543 (Springer, Berlin, 1991) 184 - 207.