Monotonic and Dual Monotonic Language Learning

Steffen Lange[1]
HTWK Leipzig
FB Mathematik und Informatik
PF 66
04251 Leipzig
Germany
steffen@informatik.th-leipzig.de

Thomas Zeugmann
Research Institute of
Fundamental Information Science
Kyushu University 33
Fukuoka 812
Japan
thomas@rifis.kyushu-u.ac.jp

Shyam Kapur
Department of Computer Science
James Cook University of North Queensland
Townsville QLD 4811
Australia
kapur@coral.cs.jcu.edu.au

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

Abstract

Monotonic and dual monotonic language learning from positive as well as from positive and negative examples is investigated. Three different notions of monotonicity are considered. Each of them reflects an alternative formalization of the requirement that the learner has to produce better and better generalizations when fed more and more data on the concept to be learned. *Strong–monotonicity* absolutely requires that *only* better and better *generalizations* be produced. *Monotonic learning* reflects the demand that for any two guesses the one output later has to be, with respect to the target language, at least as good as the earlier one. *Weak–monotonicity* is the analogue in learning theory of *cumulativity*. The corresponding three versions of dual monotonicity describe the requirement that the inference device only produces specializations that fit the target language better and better. *Dual strong–monotonic* learning generates a chain of shrinking *specializations* converging to the target language. *Dual monotonicity* describes the same goal with respect to the target language and *dual weak–monotonic* learning is the analogue of the dual of cumulativity. The power of each of these 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 is completely investigated, thereby obtaining strong hierarchies.

1. Introduction

The process of hypothesizing a general rule from 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*, a rapidly advancing science that started with the seminal papers of Solomonoff [46] and of Gold [16]. Computer scientists widely used their insight into the theory of computability to obtain a better and deeper understanding of processes performing inductive generalizations. The state of the art is excellently surveyed in Angluin and Smith [3, 4]. For more information concerning recent developments in inductive inference, the reader is referred to the annual Workshops on Computational Learning Theory, COLT (cf., e.g., Rivest et al. [44], Fulk and Case [15] and Haussler [17]), the International Workshops on Algorithmic Learning Theory, ALT (cf., e.g., Arikawa et al. [5, 6]) and the workshops on Analogical and Inductive Inference, AII (cf., e.g., Jantke [20, 23]).

The present paper deals with formal language learning in the limit. In this field, many interesting and sometimes surprising results have been elaborated within the last decades (cf., e.g., Osherson, Stob and Weinstein [42], Case [12], Fulk [14], Jain and Sharma [19]). 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 deal with, we describe the general situation investigated in language learning in the limit. The learner is provided with more and more information concerning the language to be inferred and has to produce, from time to time, a hypothesis about the phenomenon to be learned. The information given may contain only *positive examples*, i.e., eventually all strings from the language to be recognized, as well as both *positive and negative examples*, i.e., the learner is fed eventually all the strings over the underlying alphabet classified with respect to their containment in the target language. The set of all admissible hypotheses is called the *space of hypotheses*, or, synonymously, the *hypothesis space*. The hypothesis space 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 learned (cf. Definitions 1 and 2). In this paper we exclusively consider learning of recursively enumerable families of uniformly recursive languages. These families are of particular interest with respect to potential applications.

Starting with Angluin's [1, 2] 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 $\mathcal{L}$. In order to avoid confusion it should be mentioned that several authors use the term "exact learning" to denote the requirement that the hypothesis the learning algorithm converges to does correctly describe the target language (cf., e.g., Osherson, Stob and Weinstein [42]). The requirement to learn a target family $\mathcal{L}$ with respect to $\mathcal{L}$ itself is sometimes also referred to as "proper learning" (cf., e.g., Haussler [17]).

With respect to potential applications the requirement to learn exactly seems to be a very natural one. A potential user of a learning system may really prefer some particular representation for all the objects to be learned. However, in most cases their are many alternative representations of the target languages. Hence, it is only natural to ask whether the demand of our hypothetical user seriously affects the capabilities of learning systems. To illustrate our point, we need some additional notation.

Let $\mathcal{L} = L_0, L_1, \ldots$ be an enumeration of a family of target languages. That is, we are given a specific description of each $L \in \mathcal{L}$, and a particular enumeration of $\mathcal{L}$ as a whole. We call a target family $\mathcal{L}$ learnable by a *class preserving* inference algorithm, if there is a hypothesis space $\mathcal{G} = G_0, G_1, \ldots$ such that every G_j describes a language that is contained in $\mathcal{L}$, and the learning algorithm infers $\mathcal{L}$ with respect to $\mathcal{G}$. Clearly, every $L \in \mathcal{L}$ has to possess a description G_j in $\mathcal{G}$; otherwise, the learning algorithm cannot infer all of $\mathcal{L}$ with respect to $\mathcal{G}$. Notice that when dealing with class preserving language learning, in contrast to exact learning, we are not only free to choose a *different enumeration* of $\mathcal{L}$ but also *alternative descriptions* for the target languages $L \in \mathcal{L}$. This approach may considerably increase the learning capabilities, as we shall see (cf., e.g., Theorem 3). At first glance, class preserving learning does not seem to be constraining, since any hypothesis that does not describe a language from the target family $\mathcal{L}$ is definitely wrong. And indeed, as long as learning in the limit as described above is concerned, this impression is true (cf. Proposition 1). However, results have been obtained which show that at least the efficiency of learning may be seriously affected if one insists on class preserving inference (cf., e.g., Pitt and Valiant [43], Blum and Singh [10], Lange and Zeugmann [33], Lange [29]). Therefore, we consider *class comprising* learning, too. In this setting a learning algorithm is allowed to use any hypothesis space $\mathcal{G} = G_0, G_1, \ldots$ such that every $L \in \mathcal{L}$ possesses a description in $\mathcal{G}$ but $\mathcal{G}$ may additionally contain elements G_k not describing any language from $\mathcal{L}$. As we shall see, class comprising learning has its peculiarities (cf. Section 6).

The main goal of this paper consists in partially answering the question to what extend *non-monotonic reasoning* has to be incorporated into the learning process. This line of research has been initiated by Jantke [21, 22]. In classical logic it is always required that an enlargement of a set of assumptions does not lead to a decrease of derivable theorems. This is the so-called *monotonicity principle.* Adapting this principle to learning leads to the following inference model. We interpret all the data a learning device is fed as assumptions concerning the target language. Each hypothesis is then just describing a set of conclusion derived from the input data. Hence, interpreting the monotonicity principle in its *strongest* sense directly leads to the requirement that the learning algorithm is forced to output an augmenting chain of languages, i.e., $L_i \subseteq L_j$ in case L_j is hypothesized later than L_i (cf. Definition 4 (A)). We call the resulting learning model *strong-monotonic* inference.

However, classic logic turned out to be inappropriate to model common sense (cf., e.g., Brewka [11]). Hence, it is only natural to consider weakened versions of strong-monotonic learning, too. Wiehagen [50] refined Jantke's [21] approach by relating “augmenting chain” to the target language L, and required $L_i \cap L \subseteq L_j \cap L$ provided L_j appears later in the sequence of hypotheses than L_i does (cf. Definition 4 (B)). This demand has the following more intuitive meaning: A new hypothesis is never allowed to reject some string that a previously generated guess already *correctly includes.* Learning devices behaving thus are called *monotonic*, since they are intuitively still closely related to the monotonicity principle.

The next weakening of strong-monotonic learning has a direct analogue in non-monotonic logic, i.e., *cumulativity* and goes back to Jantke [21]. A logic is called cumulative if for all sets X, Y of assumptions the following condition is fulfilled: $X \subseteq Y \subseteq C(X)$ implies $C(X) \subseteq C(Y)$, where $C(X), C(Y)$ denote the set of conclusions derivable from X and Y, respectively. That means, as long as no new assumption does contradict any of the conclusions derived from X the monotonicity principle holds. On the other hand, if $Y \not\subseteq C(X)$ then there are no requirements at all concerning $C(Y)$. Adapting cumulativity to learning directly results in demanding a learner to behave strong-monotonically as long as it does not receive data contradicting its actual hypothesis. If the actual guess is contradicted then the learner is allowed to output any new hypothesis (cf. Definition 4 (C)).

All the models of monotonic learning introduced above additionally share the property that they model "learning by generalization." Hence, it is only natural to consider the dual counterparts of them, i.e., "learning by specialization." This approach might lead to learning algorithms that initially output a most general hypothesis. Afterwards, the inference device might need to specialize its hypotheses until it eventually reaches a correct guess. It is also plausible that these two strategies might have to be combined to learn, that is, a suitable interplay between generalization and specialization is essential in order to reach the learning goal. Formalizing the latter approaches directly yield *dual strong-monotonic, dual monotonic* and *dual weak-monotonic* learning (cf. Definition 6). There has been an intensive debate in the machine learning community for and against each of these learning models (cf., e.g., [28], [39], [40]). Hence, our results may be also regarded as a partial answer to the problem what is more appropriate, learning by specialization, generalization or the combination of both.

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 (cf. Corollary 11). However, 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.

On the other hand, the situation considerably changes if learning from *positive* data is concerned. First of all, the learning capabilities of the different models of monotonic and dual monotonic learning *crucially depend* on the appropriate choice of the relevant hypothesis space. As long as class comprising learning is considered, dual weak monotonic learning is still as *powerful* as learning in the limit (cf. Theorem 27). However, class preserving dual weak-monotonic learning already results in a severe restriction of the learning capabilities (cf. Theorem 9). Moreover, all other notions of monotonicity seriously shrink the learning power.

The paper is structured as follows. Section 2 presents preliminaries, i.e., notations, definitions and motivations. In Section 3 we exclusively deal with learning from positive data. The learnability from both positive and negative data is studied in Section 4. Moreover, this section also answers the question whether or not information presentation can be traded versus monotonicity constraints. Section 5 is devoted to the problem whether or not "natural families" of formal languages may be inferred under some of the monotonicity constraints described above. Finally, we give a summary and present open problems. All references are listed in Section 7.

2. Preliminaries

Let $\mathbb{N} = \{0, 1, 2, ...\}$ be the set of all natural numbers. We set $\mathbb{N}^+ = \mathbb{N} \setminus \{0\}$. Let $\varphi_0,\ \varphi_1,\ \varphi_2, ...$ denote any fixed ***programming system*** of all (and only all) partial recursive functions over $\mathbb{N}$, and let $\Phi_0,\ \Phi_1,\ \Phi_2, ...$ be any associated ***complexity measure*** (cf. Machtey and Young [38]). Then φ_k is the partial recursive function computed by program k in the programming system. Furthermore, let $k, x \in \mathbb{N}$. If $\varphi_k(x)$ is defined (abbr. $\varphi_k(x) \downarrow$) then we also say that $\varphi_k(x)$ converges; otherwise, $\varphi_k(x)$ diverges (abbr. $\varphi_k(x) \uparrow$). By $\langle .,. \rangle : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ we denote ***Cantor's pairing function***, i.e., $\langle x, y \rangle = ((x+y)^2 + 3x + y)/2$ for all $x, y \in \mathbb{N}$.

In the sequel we assume familiarity with formal language theory (cf., e.g., Hopcroft and Ullman [18]). 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 let $t = s_0, s_1, s_2, ...$ be an infinite sequence of strings from Σ^* such that $range(t) = \{s_k \,|\, k \in \mathbb{N}\} = L$. Then t is said to be a ***text*** for L or, synonymously, a ***positive presentation***. Let L be a language. By $text(L)$ we denote the set of all positive presentations of L. Furthermore, let $i = (s_0, b_0), (s_1, b_1), ...$ be an infinite sequence of elements of $\Sigma^* \times \{+, -\}$ such that $range(i) = \{s_k \,|\, k \in \mathbb{N}\} = \Sigma^*$, $i^+ = \{s_k \,|\, (s_k, b_k) = (s_k, +), k \in \mathbb{N}\} = L$ and $i^- = \{s_k \,|\, (s_k, b_k) = (s_k, -), k \in \mathbb{N}\} = co-L$. 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***. Let L be a language. By $info(L)$ we denote the set of all informants for L. 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 $x+1$, respectively, e.g., $i_2 = (s_0, b_0), (s_1, b_1), (s_2, b_2)$. Let t be a text and let $x \in \mathbb{N}$. Then we define $t_x^+ = \{s_k \,|\, k \le x\}$. Furthermore, by i_x^+ and i_x^- we denote the sets $\{s_k \,|\, (s_k, +) \in i, k \le x\}$ and $\{s_k \,|\, (s_k, -) \in i, k \le x\}$, respectively. Finally, we write $t_x \sqsubseteq t_y$ ($t_x \sqsubset t_y$), iff t_x is a (proper) prefix of t_y.

Next we introduce the notion of the ***canonical text*** that will be very helpful in proving several theorems. Let L be any non-empty recursive language, and let $s_0, s_1, ...$ be the lexicographically ordered text of Σ^*. Test sequentially whether $s_z \in L$, for $z = 0, 1, 2, ...,$ until the first z is found such that $s_z \in L$. Since $L \neq \emptyset$, there must be at least one z fulfilling the test. Set $t_0 = s_z$. We proceed inductively, $x \ge 0$,

$$t_{x+1} = \begin{cases} t_x \cdot s_{z+x+1}, & \text{if} \quad s_{z+x+1} \in L, \\ t_x \cdot s, & \text{otherwise, where } s \text{ is the last string in } t_x. \end{cases}$$

Following Angluin [1], we restrict ourselves to deal exclusively with indexed families of uniformly recursive languages defined as follows: A sequence $L_0, L_1, L_2, ...$ is said to be an ***indexed family*** $\mathcal{L}$ of uniformly 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

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

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 uniformly recursive languages (cf. Hopcroft and Ullman [18]). In the following we refer to indexed families of uniformly recursive languages as indexed families for short. Moreover, we sometimes denote an indexed family and its range by the same symbol $\mathcal{L}$. The meaning will be clear from the context.

As in Gold [16], 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 (or an informant i) and it either requests the next input, or it first outputs a hypothesis, i.e., a number encoding a certain computer program, and then it requests the next input (cf., e.g., Angluin [1]).

At this point we have to clarify what hypothesis space we should choose, thereby also specifying the goal of the learning process. Gold [16] and Wiehagen [49] pointed out that there is a difference in what can be inferred depending 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 [13] 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. 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 learned, i.e., to require containment of languages as described in the introduction. It turns out that this latter approach considerably increases the power of all types of monotonic and dual monotonic language learning. Furthermore, since we exclusively deal with the learnability of indexed families $\mathcal{L} = (L_j)_{j\in\mathbb{N}}$ we always take as the hypothesis space an enumerable family of grammars $G_0, G_1, G_2, ...$ over the terminal alphabet Σ satisfying $\mathcal{L} \subseteq \{L(G_j) \,|\, j \in \mathbb{N}\}$. Moreover, we require that membership in $L(G_j)$ is uniformly decidable for all $j \in \mathbb{N}$ 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. When an IIM outputs a number j, we interpret it to mean that the machine is hypothesizing the grammar G_j.

Let σ be a text or informant, respectively, and $x \in \mathbb{N}$. Then we use $M(\sigma_x)$ to denote the last hypothesis produced by M when successively fed σ_x. The sequence $(M(\sigma_x))_{x\in\mathbb{N}}$ is said to ***converge in the limit*** to the number j if and only if either $(M(\sigma_x))_{x\in\mathbb{N}}$ is infinite and all but finitely many terms of it are equal to j, or $(M(\sigma_x))_{x\in\mathbb{N}}$ is non-empty and finite, and its last term is j. Now we are ready to define learning in the limit.

Definition 1. (Gold, [16]) *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ be a hypothesis space.* ***An IIM M CLIM–TXT [CLIM–INF]–identifies L from***

text [***informant***] ***with respect to $\mathcal{G}$*** *iff for every text t [informant i] for L, there exists a $j \in \mathbb{N}$ such that the sequence $(M(t_x))_{x\in\mathbb{N}}$ $[(M(i_x))_{x\in\mathbb{N}}]$ converges in the limit to j and $L = L(G_j)$.*

Furthermore, M $CLIM-TXT$ $[CLIM-INF]$–identifies $\mathcal{L}$ with respect to $\mathcal{G}$ iff, for each $L \in \mathcal{L}$, M $CLIM-TXT$ $[CLIM-INF]$–identifies L with respect to $\mathcal{G}$.

Finally, let $CLIM-TXT$ $[CLIM-INF]$ denote the collection of all indexed families $\mathcal{L}$ for which there is an IIM M and a hypothesis space $\mathcal{G}$ such that M $CLIM-TXT$ $[CLIM-INF]$–identifies $\mathcal{L}$ with respect to $\mathcal{G}$.

Since, by the definition of convergence, only finitely many data of L were seen by the IIM upto the (unknown) point of convergence, whenever an IIM identifies the language L, some form of learning must have taken place. For this reason, hereinafter the terms *infer*, *learn*, and *identify* are used interchangeably.

In Definition 1, LIM stands for "limit." Furthermore, the prefix C is used to indicate ***class comprising*** learning, i.e., the fact that $\mathcal{L}$ may be learned with respect to some hypothesis space comprising $range(\mathcal{L})$. The restriction of $CLIM$ to ***class preserving*** inference is denoted by LIM. That means LIM is the collection of all indexed families $\mathcal{L}$ that can be learned in the limit with respect to a hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ such that $range(\mathcal{L}) = \{L(G_j) \mid j \in \mathbb{N}\}$. Moreover, if a target indexed family $\mathcal{L}$ has to be inferred with respect to the hypothesis space $\mathcal{L}$ itself, then we replace the prefix C by E, i.e., $ELIM$ is the collection of indexed families that can be ***exactly*** learned in the limit. We adopt this convention in the definitions of the learning types below.

Definition 1 could be easily generalized to arbitrary families of recursively enumerable languages (cf., e.g., Osherson et al. [42]). Nevertheless, we exclusively consider the restricted case defined above, since our motivating examples are all families of uniformly recursive languages. Furthermore, the following question naturally arises. Does the collection of inferable indexed families depend on the set of allowed hypothesis spaces introduced above? We answer this question with the following proposition that completely clarifies the relations between exact, class preserving and class comprising learning in the limit.

Proposition 1. (Lange and Zeugmann, [33])

(1) $ELIM-TXT = LIM-TXT = CLIM-TXT$

(2) $ELIM-INF = LIM-INF = CLIM-INF$

Note that, in general, it is not decidable whether or not M has already inferred L. With the next definition, we consider a special case where it is required that the IIM declares that it has finished the learning task.

Definition 2. (Gold, [16]; Trakhtenbrot and Barzdin, [47]) *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ be a hypothesis space.* ***An IIM M CFIN–TXT*** [***CFIN–INF***]–***identifies L from text*** [***informant***] ***with respect to $\mathcal{G}$*** *iff for every text t [informant i] for L, there exists a $j \in \mathbb{N}$ such that M, when successively fed t [i], outputs the single hypothesis j, $L = L(G_j)$, and stops thereafter.*

Furthermore, M $CFIN-TXT$ [$CFIN-INF$]–identifies $\mathcal{L}$ with respect to $\mathcal{G}$ iff, for each $L \in \mathcal{L}$, M $CFIN-TXT$ [$CFIN-INF$]–identifies L with respect to $\mathcal{G}$.

The resulting learning type is denoted by $CFIN-TXT$ $[CFIN-INF]$.

Next, we want to formally define strong-monotonic, monotonic and weak-monotonic inference. But before doing this, we first define *consistent* identification. Consistent learning devices have been introduced by Barzdin (1974) as well as by Blum and Blum (1975). Intuitively, consistency means that the IIM has to correctly reflect the information it has already been fed.

Definition 3. (Barzdin, [7]; Blum and Blum, [9]) *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ be a hypothesis space.* ***An IIM M CCONS–TXT [CCONS–INF]–identifies L from text [informant] with respect to $\mathcal{G}$*** *iff*

(1) *M $CLIM-TXT$ [$CLIM-INF$]–infers L from text [informant] with respect to $\mathcal{G}$,*

(2) *for every text t [informant i] for L the following condition is satisfied: whenever M on t_x [on i_x] produces a hypothesis j_x, then $t_x^+ \subseteq L(G_{j_x})$ [$i_x^+ \subseteq L(G_{j_x})$ and $i_x^- \subseteq co{-}L(G_{j_x})$].*

Moreover, M $CCONS-TXT$ [$CCONS-INF$]–identifies $\mathcal{L}$ with respect to $\mathcal{G}$ iff, for each $L \in \mathcal{L}$, M $CCONS-TXT$ [$CCONS-INF$]–identifies L with respect to $\mathcal{G}$.

$CCONS-TXT$ and $CCONS-INF$ are analogously defined as above.

Now we are ready to formally define the three types of monotonic language learning introduced in Section 1.

Definition 4. (Jantke, [21], Wiehagen, [50]) *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ be a hypothesis space.* ***An IIM M is said to identify a language L from text [informant] with respect to $\mathcal{G}$***

(A) ***strong-monotonically***

(B) ***monotonically***

(C) ***weak-monotonically***

iff

M $CLIM-TXT$ [$CLIM-INF$]–infers L with respect to $\mathcal{G}$ and for every 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}], respectively, where $k \geq 1, k \in \mathbb{N}$, 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 $L(G_{j_x}) \subseteq L(G_{j_{x+k}})$].*

In particular, Requirement (C) means that M has to work strong-monotonically as long as its guess j_x is consistent with *all* the data fed to M both before and after M has output j_x.

We denote by $CSMON-TXT$, $CSMON-INF$, $CMON-TXT$, $CMON-INF$, $CWMON-TXT$, $CWMON-INF$ the collection of all thoses indexed families $\mathcal{L}$ for which there is a hypothesis space $\mathcal{G}$ and an IIM inferring them strong-monotonically, monotonically, and weak-monotonically from text or informant with respect to $\mathcal{G}$, respectively.

This is a good place to give an example showing the inferring power of our approach.

Example 1. Let $L \subseteq \Sigma^*$ be some arbitrarily chosen *infinite* context-sensitive language. By $\mathcal{L}_{fin}$ we denote the set of all finite languages over Σ. We define $\mathcal{L}_{finvar}$ to be the canonical enumeration of $\{L \cup L_{fin} \mid L_{fin} \in \mathcal{L}_{fin}\}$. Clearly, $\mathcal{L}_{finvar}$ constitutes an indexed family. In case we would require the IIMs to output programs of characteristic functions and interpret the monotonicity requirements with respect to the graphs of these functions, we immediately obtain from Jantke [21] that, even from informant, $\mathcal{L}_{finvar}$ cannot be learned strong-monotonically. On the other hand, we now show that in our approach $\mathcal{L}_{finvar}$ is strong-monotonically learnable with respect to $\mathcal{L}_{finvar}$ from *text*. Let $L \in \mathcal{L}$, and $t \in text(L)$. The IIM M performing the inference process initially outputs the canonical index 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 or not the new string s belongs to L_{j_x}. In case it does, let $j_{x+1} = j_x$. Otherwise, M outputs the canonical number j_{x+1} of $L_{j_x} \cup \{s\}$. 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 is indeed a strong-monotonic IIM. $\diamond$

Next, we define *conservative* IIMs. Intuitively speaking, conservative IIMs maintain their actual hypothesis at least as long as they have not seen data contradicting it. Hence, whenever a conservative IIM performs a mind change it is because it has perceived clear inconsistency between its guess and the input.

Definition 5. (Angluin, [1]) *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space.* ***An IIM M CCONSERVATIVE–TXT [CCONSERVATIVE–INF]–infers L from text [informant] with respect to $\mathcal{G}$*** *iff*

(1) *M $CLIM-TXT$ $[CLIM-INF]$–identifies L with respect to $\mathcal{G}$,*

(2) *for every text t [informant i] for L the following condition is satisfied: whenever M on input t_x [on i_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 either fail to contain some string $s \in i_{x+k}^+$ or it generates some string $s \in i_{x+k}^-$].*

Finally, M CCONSERVATIVE–TXT [CCONSERVATIVE–INF]–identifies $\mathcal{L}$ with respect to $\mathcal{G}$ if and only if, for each $L \in \mathcal{L}$, M CCONSERVATIVE–TXT [CCONSERVATIVE–INF]–identifies L with respect to $\mathcal{G}$.

The collection of sets *CCONSERVATIVE–TXT* and *CCONSERVATIVE–INF* are defined in a manner analogous to that above.

We continue in formally defining the three types of dual monotonic language learning introduced in Section 1.

Definition 6. *Let $\mathcal{L}$ be an indexed family, $L \in \mathcal{L}$, and let $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$ be a hypothesis space.* ***An IIM M is said to identify a language L from text [informant] with respect to $\mathcal{G}$***

(A) ***dual strong-monotonically***

(B) ***dual monotonically***

(C) ***dual weak-monotonically***

iff

M $CLIM-TXT$ $[CLIM-INF]$–identifies L with respect to $\mathcal{G}$ 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}], respectively, for some $k \geq 1, k \in \mathbb{N}$, 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 $CSMON^d{-}TXT$, $CSMON^d{-}INF$, $CMON^d{-}TXT$, $CMON^d{-}INF$, $CWMON^d{-}TXT$, and $CWMON^d{-}INF$ we denote the collections of all those indexed families $\mathcal{L}$ for which there is a hypothesis space $\mathcal{G}$ and an IIM identifying them dual strong-monotonically, dual monotonically and dual weak-monotonically from text or informant with respect to $\mathcal{G}$, respectively.

Note that the matching notions of monotonicity and of dual monotonicity are truly duals of *each other*. Next, we combine the monotonicity constraints from Definition 4 and Definition 6. As we shall see, this helps us appreciate better relationships between monotonic inference of language families and other well-known types of language learning.

Definition 7. *Let $CSMON^{\&}-TXT$ $[CSMON^{\&}-INF]$ denote the class of indexed families learnable by an IIM that works strong-monotonically as well as dual strong-monotonically. The learning types $CMON^{\&}{-}TXT$, $CMON^{\&}{-}INF$, $CWMON^{\&}{-}TXT$, and $CWMON^{\&}-INF$ are defined analogously.*

We may also want to consider combinations of different requirements, say, for example, strong-monotonicity and consistency. To satisfy this combined requirement, an IIM would have to perform the inferring process in both strong-monotonic *and* consistent fashion. The corresponding class of learnable indexed families is denoted by $CSMON{-}CONS{-}TXT$ ($CSMON{-}CONS{-}INF$).

In the next section, we present a variety of results concerning text identification. Note that '$\subset$' denotes proper set inclusion and '#' denotes incomparability of sets.

3. Monotonic inference from text

In this section, we systematically investigate the relations between the different models of monotonic and dual monotonic language learning from text. Thereby, we mainly focus our attention on class preserving learning. This provides us a deeper understanding of the strengths as well as of the limitations of monotonic and dual monotonic learning.

We start with an example which points to a particular weakness of strong-monotonic learning. Namely, if a learnable indexed family contains an accumulation point (cf. Kapur [25]) then strong-monotonic inference is equivalent to finite learning.

Example 2. Consider the following indexed family $\mathcal{L}_{mon} = (L_k)_{k\in\mathbb{N}}$. We set $L_0 = \{a\}^+$ and for all $k \in \mathbb{N}^+$, $L_k = \{a^z \mid 1 \leq z \leq k\} \cup \{b^k\}$. Let $\mathcal{L}_{mon} = (L_k)_{k\in\mathbb{N}}$. Even if positive and negative examples are provided, there does not exists any strong-monotonic IIM M which learns $\mathcal{L}_{mon}$. The main obstacle is that any initial segment of an informant for L_0 may serve as an initial segment of an informant for infinitely many languages L_k provided that k is sufficiently large. Since all languages in $\mathcal{L}_{mon}$ are pairwise disjoint, M cannot handle this situation successfully (cf. the proof of Theorem 1). $\diamond$

However, in general strong-monotonic inference is more powerful than finite learning (cf. Theorem 2 and Theorem 13).

Next, we compare the learning power of the different models of class preserving monotonic inference. Our first theorem actually shows that there is a strong hierarchy between the different notions of monotonicity. The corresponding learning types can be separated by using quite simple families of regular languages.

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 indexed family $\mathcal{L}_{mon}$ from Example 2.

First, we present an IIM M monotonically inferring $\mathcal{L}_{mon}$ from text with respect to $\mathcal{L}_{mon}$. Let $L \in \mathcal{L}$, and let $t \in text(L)$. As long as the part of t fed to M only contains strings over the alphabet $\{a\}$, the IIM M outputs the hypothesis 0. In case a string of the form b^k appears in the text, M outputs the guess k. This hypothesis remains unchanged in every subsequent step.

Obviously, M infers every $L \in \mathcal{L}_{mon}$. Thereby, M changes its mind at most one time, say from 0 to k. Clearly, $L_0 \cap L_k \subseteq L_k$. Thus, M is a monotonic IIM. This proves the first part.

It remains to show that there is no IIM identifying $\mathcal{L}_{mon}$ strong-monotonically from text. As already announced in Example 2, we prove the stronger result that $\mathcal{L}_{mon} \notin SMON{-}INF$.

Suppose the converse, that is, assume there is an IIM M and a class preserving hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ such that M strong-monotonically learns $\mathcal{L}_{mon}$ from informant with respect to $\mathcal{G}$. Let $i \in info(L_0)$. Then there must be an x such that M after having fed i_x outputs a grammar for L_0 because otherwise M cannot infer L_0. Now, let k be any fixed number satisfying $k > max\{|s| \mid s \in i_x^+ \cup i_x^-\}$. Then consider i_x and $i_x, (b^k, +)$, two initial segments of an informant for L_k. Due to our choice of i_x, M first outputs j_x such that $L(G_{j_x}) = L_0$. Let $\hat{\imath}$ be any informant for L_k such that $i_x, (b^k, +)$ is a prefix of $\hat{\imath}$. Clearly, there has to be a $y \geq x + 1$ satisfying $M(\hat{\imath}_y) = j_y$, $L_k = L(G_{j_y})$, since otherwise M cannot infer L_k. However, $L_0 \not\subseteq L_k$, and hence M violates the strong-monotonicity constraint. This completes the proof.

Next we show $MON{-}TXT \subseteq WMON{-}TXT$. The idea is as follows. Suppose M monotonically learns $\mathcal{L}$ with respect to a class preserving hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$. We construct a machine $\hat{M}$ that weak-monotonically infers $\mathcal{L}$ with respect to the same hypothesis space $\mathcal{G}$. The machine $\hat{M}$ simulates M. On every text, it initially guesses whatever M guesses. Thereafter, its behavior is conservative. If, at any stage, it finds some evidence that contradicts its current guess, it adopts whatever M's guess is.

Clearly, the only way $\hat{M}$ can fail to converge to the target language L is if, at some point, it guesses some superset of L. Since the guesses of $\hat{M}$ are a subset of M's guesses, this reduces to the question whether or not M may eventually output an overgeneralized hypothesis j, that is, $L \subset L(G_j)$. However, this is impossible, since otherwise M can be proved to violate the monotonicity demand. The formal proof is done using the following claim:

Claim. Let $\mathcal{L}$ be any indexed family, and let M be any monotonic IIM inferring $\mathcal{L}$ with respect to some hypothesis space $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$. Let $L \in \mathcal{L}$, and $t \in \mathit{text}(L)$. 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, that is, 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 can be extended to 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 extension of t_{x+k}, say $\hat{t}_{x+k+z}$, with strings from $L(G_j)$ such that $j_{x+k+z} := M(\hat{t}_{x+k+z})$ and $L(G_{j_{x+k+z}}) = L(G_j)$.

Since $L(G_j) \supset L = L(G_{j_{x+k}}) \cap L(G_j) \subset L(G_{j_{x+k+z}}) \cap L(G_j) = L(G_j)$ we have found a text $\hat{t}$ 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 $\mathcal{G}$ to be a class preserving hypothesis space, M exclusively guesses languages belonging to $\mathcal{L}$. Thus, $\mathcal{L} \in WMON{-}TXT$ is witnessed by $\hat{M}$.

For the remaining part, that is, $WMON{-}TXT \setminus MON{-}TXT \neq \emptyset$, the reader is referred to the proof of Theorem 16, Claim B and C, where the stronger result $WMON{-}TXT \setminus MON{-}INF \neq \emptyset$ is shown. □

At this point it is natural to ask what the lower and upper bounds of this hierarchy are. The answer is provided by the next theorem.

Theorem 2.

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

(2) $WMON{-}TXT = CONSERVATIVE{-}TXT$

Proof. By definition, if M finitely infers $\mathcal{L} \in FIN{-}TXT$, then, on every text for every language in $\mathcal{L}$, M outputs only one 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 $\mathcal{L}_{finvar} = \{L \cup L_{fin} \mid L_{fin} \in \mathcal{L}_{fin}\}$, that is, the family defined in Example 1.

Since a conservative IIM exclusively performs justified mind changes, the inclusion $CONSERVATIVE{-}TXT \subseteq WMON{-}TXT$ is obvious. Now suppose M weak-monotonically learns $\mathcal{L}$ with respect to $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$. We construct a machine $\hat{M}$ that conservatively learns $\mathcal{L}$ with respect to the same hypothesis space $\mathcal{G}$. $\hat{M}$ is easily obtained from M

by simply adding a consistency test, that is, let $j_x := M(t_x)$ and $s \in t^+_{x+k}$ for some $k = 0, 1, \ldots$. Then test whether $s \in L(G_{j_x})$ for all s. In case it is, $\hat{M}$ repeats j_x as its hypothesis. Otherwise, it outputs $M(t_{x+k})$.

Obviously, if M avoids overgeneralization, then $\hat{M}$ learns $\mathcal{L}$ as required.

Claim. Let $\mathcal{L}$ be any indexed family, and let M be any weak-monotonic IIM inferring $\mathcal{L}$ with respect to some hypothesis space $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$. Let $L \in \mathcal{L}$, and let $t \in text(L)$ be arbitrarily fixed. When fed t, M never produces a guess j such that $L \subset L(G_j)$.

Suppose the converse, that is, there is an index x such that $M(t_x) = j,\ L \subset L(G_j)$. Since $L \subset L(G_j)$, we obtain $t^+_{x+r} \subseteq L(G_j)$ for every $r \in \mathbb{N}^+$. Since M is a weak-monotonic IIM, M is never allowed to output a guess z such that $L(G_z) \subset L(G_j)$. Thus, M fails to infer L from text t, a contradiction. This proves the claim.

Notice that in contrast to the proof for $MON\text{–}TXT \subseteq WMON\text{–}TXT$ above the class preserving condition on the hypothesis space is not essential. □

Next we ask whether $LIM-TXT$ and $WMON-TXT$ may be separated. For exact learning, Angluin [1] proved that conservative learning devices are less powerful than unrestricted IIMs. However, her proof technique is not applicable to the class-preserving case. We wonder whether the insistence on exact learning accounts for the considerable reduction in learning power. Our next theorem impressively establishes the veracity of our conjecture.

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

Proof. The desired indexed family is defined as follows. For all $k \in \mathbb{N}$, we set $L_{\langle k,0\rangle} = \{a^k b^m \mid m \in \mathbb{N}^+\}$. By convention, a^0 equals the empty string. Let $j > 0$. We distinguish the following cases.

Case 1. $\neg\ \Phi_k(k) \leq j$

Then we set $L_{\langle k,j\rangle} = L_{\langle k,0\rangle}$.

Case 2. $\Phi_k(k) \leq j$

We set $L_{\langle k,j\rangle} = \{a^k b^m \mid 1 \leq m \leq \Phi_k(k)\}$.

Obviously, $\mathcal{L} = (L_{\langle k,j\rangle})_{k,j \in \mathbb{N}}$ is an indexed family of recursive languages. The following observation about the family $\mathcal{L}$ is used in the proof of Claim B below: For all $k,\ j,\ r \in \mathbb{N}$, $a^k b^j \in L_{\langle k,j\rangle}$ if and only if $a^k b^j \in L_{\langle k,j+r\rangle}$.

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

Suppose the converse, that is, $\mathcal{L} \in EWMON\text{–}TXT$. We shall show that M may be used to decide the halting problem. Since the halting problem is algorithmically undecidable, no IIM M can weak-monotonically infer $\mathcal{L}$ with respect to $\mathcal{L}$. Let $k \in \mathbb{N}$, and let t be the canonical text for $L_{\langle k,0\rangle}$. Since M, in particular, infers $L_{\langle k,0\rangle}$ from t, there is an minimal z such that M on input t_z computes a consistent hypothesis. By definition of $\mathcal{L}$, we obtain $M(t_z) = \langle k,j\rangle$ for some $j \in \mathbb{N}$. Let $m = max\{z, j\}$. We claim that the following effective test determines whether or not $\varphi_k(k)\downarrow$.

If $\Phi_k(k) \leq m$ then $\varphi_k(k)\downarrow$ else $\varphi_k(k)\uparrow$.

The 'if' direction is trivial. Suppose $\varphi_k(k)\downarrow$ and the 'if' condition fails. Let $\Phi_k(k) = y$. Obviously, $y > m$. By definition of $\mathcal{L}$, $L_{\langle k,y\rangle}$ is finite. On the other hand, $y > j$ implies that $L_{\langle k,j\rangle}$ is an infinite language which satisfies $L_{\langle k,y\rangle} \subset L_{\langle k,j\rangle}$. Since $y > z$, t_z is an initial segment of the canonical text for $L_{\langle k,y\rangle}$. Hence, M has produced an overgeneralized hypothesis when inferring $L_{\langle k,y\rangle}$ from its canonical text. As in the proof of Theorem 2 one can easily argue that this contradicts our assumption that M weak-monotonically infers $L_{\langle k,y\rangle}$ from text t. Hence, no IIM can weak-monotonically infer $\mathcal{L}$ with respect to the hypothesis space $\mathcal{L}$.

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

We first define an appropriate hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$. For all $k \in \mathbb{N}$, let $L(G_{2k}) = L_{\langle k,0\rangle}$, and $L(G_{2k+1}) = \bigcap_{j\in\mathbb{N}} L_{\langle k,j\rangle}$.

By the observation made above it is easy to verify that $\mathcal{G}$ is a class preserving hypothesis space for $\mathcal{L}$.

It remains to show that there is an IIM M which strong-monotonically infers $\mathcal{L}$ with respect to the hypothesis space $\mathcal{G}$. Assume any $L \in \mathcal{L}$, any text t for L, and any $x \in \mathbb{N}$. We define the desired IIM M as follows:

IIM *M*: "On input t_x do the following: Determine the unique $k \in \mathbb{N}$ such that $t_0 = a^k b^z$ for some $z \in \mathbb{N}$. If $t_x^+ \subseteq L(G_{2k+1})$, output $2k+1$ and request the next input. Otherwise, output $2k$ and request the next input."

It is easy to see 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, $L(G_j)$ is finite and, therefore, there has to be a y such that $t_y^+ \not\subseteq L(G_j)$. Due to our definition, M changes its mind to the correct hypothesis $j-1$ which will be repeated in every subsequent step. Since $L(G_j) \subseteq L(G_{j-1})$, it follows that M strong-monotonically infers L on text t. □

Theorem 3 directly implies that exact strong-monotonic, exact monotonic and exact weak-monotonic learning are all less powerful than the corresponding class preserving models of inference. This reproves results recently obtained in Lange and Zeugmann [34].

Our next example sheds some light on the power of dual monotonic learners. The example is selected in a way such that it can be used to witness the desired separation between $LIM{-}TXT$ and $WMON{-}TXT$.

Example 3. Consider the indexed family $\mathcal{L}_{dmon}$. For all $k \in \mathbb{N}$, we set $L_{\langle k,0\rangle} = \{a^k b^n \mid n \in \mathbb{N}^+\}$. For all $k \in \mathbb{N}$ and all $j > 0$, we distinguish the following cases:

Case 1. $\neg\, \Phi_k(k) \le j$

Then we set $L_{\langle k,j\rangle} = L_{\langle k,0\rangle}$.

Case 2. $\Phi_k(k) \le j$

We distinguish the following subcases.

Subcase 2.1. $j < 2\Phi_k(k)$

Let $r = 2\Phi_k(k) - j$. We set $L_{\langle k,j\rangle} = \{a^k b^m \mid 1 \le m \le r\}$.

Subcase 2.2. $j \ge 2\Phi_k(k)$

Then we set $L_{\langle k,j\rangle} = \{a^k b\}$.

$\mathcal{L}_{dmon} = (L_{\langle k,j\rangle})_{j,k\in\mathbb{N}}$ is an indexed family of recursive languages, since the predicate "$\Phi_i(y) \le z$" is uniformly decidable for all $i, y, z \in \mathbb{N}$.

Let k, $j \in \mathbb{N}$. A dual monotonic IIM may work as follows in order to learn $L_{\langle k,j\rangle}$ from text. Let t be any text for $L_{\langle k,j\rangle}$. As long as $\varphi_k(k)\downarrow$ cannot be proved, that is, $\neg\ \Phi_k(k) \le x$ has been verified, M outputs the possibly overgeneralized hypothesis $\langle k, 0\rangle$ when fed t_x. If $\varphi_k(k)\downarrow$ has been shown, M shrinks its former guess and outputs an index of the least language L consistent with all data read so far. After this point, M behaves like a strong-monotonic learner. By definition, M is indeed a dual monotonic IIM. For the remaining details the reader is referred to the proof of Theorem 4 below. $\diamond$

As a matter of fact, the dual monotonic IIM defined above realizes a suitable interplay between learning by specialization and learning by generalization. In doing so, it uses its potential ability to output overgeneralized guesses.

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

Proof. We show that the family $\mathcal{L}_{dmon}$ introduced in Example 3 witnesses the wanted separation. Since the halting problem is undecidable, $\mathcal{L}_{dmon} \notin WMON\!-\!TXT$ follows by contraposition of the following claim.

Claim. If there exists an IIM M witnessing $\mathcal{L}_{dmon} \in WMON\!-\!TXT$, then one can effectively construct an algorithm deciding for all $k \in \mathbb{N}$ whether or not $\varphi_k(k)$ converges.

Let M be any IIM weak-monotonically inferring $\mathcal{L}_{dmon}$ with respect to some class preserving hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$. We define an algorithm $\mathcal{A}$ that solves the halting problem. On input $k \in \mathbb{N}$ the algorithm $\mathcal{A}$ executes the following instructions:

(A1) For $z = 0, 1, \ldots$ generate successively the canonical text t of $L_{\langle k,0\rangle}$ until M on input t_z outputs for the first time a hypothesis j such that $t_z^+ \cup \{a^k b^{z+2}\} \subseteq L(G_j)$.

(A2) Test whether $\Phi_k(k) \le z+1$. In case it is, output "$\varphi_k(k)$ converges" and stop. Otherwise output "$\varphi_k(k)$ diverges" and stop.

Since M has to infer $L_{\langle k,0\rangle}$ in particular from t, there has to be a least z such that M on input t_z computes a hypothesis j satisfying $t_z^+ \cup \{a^k b^{z+2}\} \subseteq L(G_j)$. Moreover, the test whether or not $t_z^+ \cup \{a^k b^{z+2}\} \subseteq L(G_j)$ can be effectively performed, since membership in $L(G_j)$ is decidable. By the definition of a complexity measure, Instruction (A2) is effectively executable. Hence, $\mathcal{A}$ is an algorithm.

It remains to show that $\mathcal{A}$ behaves as required. It suffices to show that $\varphi_k(k)$ diverges, if $\neg\ \Phi_k(k) \le z+1$. Suppose the converse; then there exists a $y > z+1$ with $\Phi_k(k) = y$. In accordance with the definition of $\mathcal{L}_{dmon}$, there is an $L \in \mathcal{L}_{dmon}$, namely $L = t_z^+$, such that $L \subset L(G_j)$. Since $t_z^+ = L$, we know that t_z is also an initial segment of a text $\hat{t}$ for L. Hence, M outputs an overgeneralized hypothesis when inferring L on $\hat{t}$. From the demonstration of Theorem 2 it follows that this contradicts our assumption that M weak-monotonically infers $\mathcal{L}_{dmon}$ with respect to the space $\mathcal{G}$. Hence, the claim is proved.

Finally, we show $\mathcal{L}_{dmon} \in EMON^d\!-\!TXT$. The IIM M defined below implements the idea already explained in Example 3. Let $L \in \mathcal{L}_{dmon}$, let $t \in text(L)$, and let $x \in \mathbb{N}$. Due to our construction, there has to be a k such that $t_x^+ \subseteq L_{\langle k,0\rangle}$. Moreover, the first string in t can be used to uniquely determine the appropriate k. We define M as follows:

IIM *M*: "On input t_x do the following: Test for each $j = 0, ..., x$, whether $a^k b^x \notin L_{\langle k,j\rangle}$. If at least one j fulfills this test, then goto (A). Otherwise, goto (B).

(A) If t_x^+ is a singleton set, then compute the minimal y such that $\{a^k b\} = L_{\langle k,y\rangle}$. Output $\langle k, y\rangle$ and request the next input.
Otherwise, compute $z = max\{n \mid t_x^+ \subseteq L_{\langle k,n\rangle}\}$. Output $\langle k, z\rangle$ and request the next input.

(B) Output $\langle k, 0\rangle$ and request the next input."

We next show that M dual monotonically infers L from text t. We distinguish between the following two cases:

Case 1. $\varphi_k(k)\uparrow$

Since $\varphi_k(k)$ is undefined, we have, for all $j \in \mathbb{N}$, $L_{\langle k,j\rangle} = L_{\langle k,0\rangle} = L$. Hence, M always produces the correct hypothesis $\langle k, 0\rangle$ in accordance with Instruction (B). Obviously, it also works dual monotonically.

Case 2. $\varphi_k(k)\downarrow$

Since $\varphi_k(k)$ is defined, there has to be a $y \in \mathbb{N}$ such that $\Phi_k(k) = y$. We now consider the following two subcases.

Subcase 2.1. $card(L) \leq y$

First, for all $z < y$, M outputs $\langle k, 0\rangle$, and then after having been fed t_y, the IIM M executes Instruction (A). There must exist some $z > y$ such that $t_z^+ = L$. It is easy to verify that $M(t_z) = \langle k, y + r\rangle$ for some $r \in \mathbb{N}$ with $L_{\langle k,y+r\rangle} = L$. Furthermore, this hypothesis will be repeated in every subsequent step. Therefore, M infers L. Finally, M works dual monotonically, since $L_{\langle k,0\rangle} \cup L \supseteq L_{\langle k,y+r+m\rangle} \cup L \supseteq ... \supseteq L_{\langle k,y+r\rangle} \cup L$. To see this, remember that $L_{\langle k,0\rangle} \supset L$ as well as, for any $m \in \mathbb{N}$, $L_{\langle k,y+r+m\rangle} \subseteq L$. Thus, applying de Morgan's rules it immediately follows that M is indeed a dual monotonic IIM.

Subcase 2.2. $card(L) = \infty$

Hence, $L = L_{\langle k,0\rangle}$. Obviously, there has to be a $z > y$ such that $t_z^+ \setminus L_{\langle k,y\rangle} \neq \emptyset$. Since all finite languages containing $a^k b$ in $\mathcal{L}_{dmon}$ are subsets of $L_{\langle k,y\rangle}$, we obtain $t_z^+ \setminus L_{\langle k,n\rangle} \neq \emptyset$ for all $n \geq y$. Since $L_{\langle k,y-1\rangle} = L_{\langle k,0\rangle} = L$, we have $M(t_z) = \langle k, y - 1\rangle$. Furthermore, this hypothesis will be repeated in every subsequent step. Finally, M works dual monotonically, since $L \supseteq L_{\langle k,j\rangle}$ for any hypothesis $\langle k, j\rangle$ produced.

Hence, M dual monotonically infers L from text t. □

As an immediate consequence of Theorem 4, we obtain the previously declared 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 analogues of Theorem 1 and 2 are valid for dual monotonic inference. The answer is given by the following theorem.

Theorem 6.

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

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

Proof. Trivially, $FIN{-}TXT \subseteq SMON^d{-}TXT$. Further, if an IIM M inferring any family dual strong-monotonically shrinks its guess, the previous guess cannot be learned dual strong-monotonically from text. As long as class preserving learning is required, M can only output equivalent hypotheses, that is, if it outputs j and subsequently k, then $L(G_j)$ must be identical to $L(G_k)$. Consequently, M's first guess has to be correct. Notice that the we have again widely stressed the class preserving demand on the hypothesis space.

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 hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$. Suppose, M does not learn $\mathcal{L}$ dual weak-monotonically. 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 to form a text for $L(G_k)$, the machine cannot revert to guessing $L(G_k)$ without violating dual monotonicity. We omit the details. Since $MON{-}TXT \subset CONSERVATIVE\text{-}TXT$ (cf. Theorem 1 and 2), the proper inclusion is an immediate consequence of both Assertion (2) of this theorem and Assertion (2) of Theorem 7 below.

To assert (2), first observe that $CONSERVATIVE{-}TXT \subseteq WMON^d{-}TXT$. It thus suffices to show $WMON^d{-}TXT \setminus CONSERVATIVE - TXT \neq \emptyset$. This last statement directly follows from Theorem 4. This completes the proof of the theorem. □

The next theorem offers deeper insight into the inference capabilities of dual monotonic IIMs both by deriving lower and upper bounds of the hierarchy and establishing links with the hierarchy involving monotonic learning classes.

Theorem 7.

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

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

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

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

Claim A. 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 $t \in text(L)$, and let $(j_x)_{x\in\mathbb{N}}$ be the sequence of M's hypotheses when successively fed 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 $r \in \mathbb{N}$. 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 as well.

Therefore, $SMON-TXT \subseteq MON^{\&}-TXT$. The proper inclusion is established by considering the family $\mathcal{L}_{mon}$ introduced in the proof of Theorem 1. Previously, we have shown that the IIM M presented in the proof of Theorem 1 behaves monotonically. It is easy to argue that it is also a dual monotonic IIM.

We proceed to prove Assertion (2). Since we already know that $MON^d-TXT \setminus MON-TXT \neq \emptyset$ (cf. Theorem 4) it remains to prove $MON-TXT \setminus MON^d-TXT \neq \emptyset$. In fact, we demonstrate a stronger result, that is, $EMON-TXT \setminus MON^d-INF \neq \emptyset$. Let $\Sigma = \{a, b, c\}$. We set $L_0 = \{a\}^+$, and, for $k \geq 1$, 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 $m > k \geq 1$. Define $\mathcal{L}$ to be the canonical enumeration of all these languages: L_k, $k \in \mathbb{N}$, and $L_{k,m}$, $k, m \in \mathbb{N}$ and $m > k$.

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

The desired IIM M which monotonically identifies $\mathcal{L}$ from text works as follows: Let $L \in \mathcal{L}$, let $t \in text(L)$, and let $x \in \mathbb{N}$. As long as $t_x^+ \subseteq \{a\}^+$, the machine outputs the canonical index for L_0. In case the first string s over $\{b\}$ does appear in the text and $t_x^+ \subseteq \{a\}^+ \cup \{b\}^+$, the IIM computes $k = min\{z \mid b^z \in t_x^+\}$ and outputs the canonical index for L_{k-1}. Finally, if a string c^m is presented, M calculates $k = min\{z \mid b^z \in t_x^+\}$ and outputs the canonical index for $L_{k-1,m}$. It is straightforward to verify that M $EMON-TXT$-identifies $\mathcal{L}$.

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

Suppose there are a hypothesis space $\mathcal{G} = (G_j)_{j \in \mathbb{N}}$ and an IIM M which dual monotonically infers $\mathcal{L}$ with respect to $\mathcal{G}$ from 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 a y such that $j_y = M(\tilde{i}_y)$ and $L(G_{j_y}) = L_k$. Let $m = max\{|s| \mid s \in \tilde{i}_y^+ \cup \tilde{i}_y^-\}$. 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 $L_{k,m}$. This can be seen as follows. When processing i_{fool}, M produces the hypothesis j_x, and then j_y. Since $b^{m+1} \in co-L_1 \cap co-L_{k,m}$ but $b^{m+1} \notin co-L_k \cap co-L_{k,m}$, M violates the dual monotonicity requirement. Hence, Assertion (2) is established.

Finally, we show (3). The part $MON^d-TXT \setminus CONSERVATIVE-TXT \neq \emptyset$ has been proved above (cf. the proof of Theorem 4). The remaining part $CONSERVATIVE-TXT \setminus MON^d-TXT \neq \emptyset$ is an immediate consequence of Claim C above. This proves the theorem. □

Notice the interesting 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 can derive the following corollary.

Corollary 8.

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

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

Finally, we investigate the relation between dual weak-monotonic learning and learning in the limit. By definition $WMON^d\text{–}TXT \subseteq LIM\text{–}TXT$. The more pretentious question is whether or not this inclusion is proper. Recently, Lange and Zeugmann [34] have shown that, in fact, $WMON^d\text{–}TXT \subset LIM\text{–}TXT$. This proof has its peculiarities distinguishing it from the proofs already given.

At first glance, the problem we are confronted with seems to be quite similar to the question how to fool a weak-monotonic learner. In both cases, it suffices to force an IIM to perform a mind change which cannot by justified by an inconsistency. But in the case of dual weak-monotonic inference, the situation is much more subtle. Namely, exclusively forcing an IIM M to output an overgeneralized hypothesis does not help, since M is allowed to specialize its guess. Hence, we have to refine both the underlying topology of the corresponding target indexed family and the related idea to fool every dual weak-monotonic learner. We achieved the desired refinement by incorporating a second halting problem into the definition of the finite languages of the indexed family introduced in Example 3.

Theorem 9. $WMON^d\!-\!TXT \subset ELIM\!-\!TXT$

Proof. We prove the theorem via the following indexed family. For all $k_1, k_2 \in \mathbb{N}$, we set $L_{\langle k_1,k_2,0\rangle} = \{a^{\langle k_1,k_2\rangle} b^z \mid z \in \mathbb{N}^+\}$. For $j > 0$, we define $L_{\langle k_1,k_2,j\rangle}$ as follows:

Case 1. $\neg\, \Phi_{k_1}(k_1) \le j$

We set $L_{\langle k_1,k_2,j\rangle} = L_{\langle k_1,k_2,0\rangle}$.

Case 2. $\Phi_{k_1}(k_1) \le j$

We distinguish the following subcases.

Subcase 2.1. $j < 2\Phi_{k_1}(k_1)$

Let $r = 2\Phi_{k_1}(k_1) - j$. We set $L_{\langle k_1,k_2,j\rangle} = \{a^{\langle k_1,k_2\rangle} b^m \mid 1 \le m \le r\} \cup \hat{L}_{\langle k_1,k_2,r\rangle}$.

Subcase 2.2. $j \ge 2\Phi_{k_1}(k_1)$

Then we set $L_{\langle k_1,k_2,j\rangle} = \{a^{\langle k_1,k_2\rangle} b\} \cup \hat{L}_{\langle k_1,k_2,1\rangle}$.

The remaining languages $\hat{L}_{\langle k_1,k_2,r\rangle}$ $(1 \le r \le \Phi_{k_1}(k_1))$ are defined as follows:

$$\hat{L}_{\langle k_1,k_2,r\rangle} = \begin{cases} \emptyset, & \text{if } \varphi_{k_2}(k_2)\uparrow, \\ \{a^{\langle k_1,k_2\rangle} c^{m+r}\}, & \text{if } \Phi_{k_2}(k_2) = m. \end{cases}$$

After a bit of reflection it is easy to see that $\mathcal{L} = (L_{\langle k_1,k_2,j\rangle})_{k_1,k_2,j\in\mathbb{N}}$ is an indexed family. First of all we show that $\mathcal{L}$ can be identified from text.

***Lemma* 1.** $\mathcal{L} \in ELIM\!-\!TXT$

Proof. Let $L \in \mathcal{L}$, $t \in \textit{text}(L)$ and let $x \in \mathbb{N}$. We define an IIM M which infers $\mathcal{L}$ with respect to the hypothesis space $\mathcal{L}$.

IIM *M*: "On input t_x do the following: If $a^{\langle k_1,k_2\rangle} c^m \in t_x^+$ for some $m \in \mathbb{N}$, then goto (A). Else, execute instruction (B).

(A) Determine the least index z such that $a^{\langle k_1,k_2\rangle}c^m \in L_{\langle k_1,k_2,z\rangle}$. Output $\langle k_1,k_2,z\rangle$ and request the next input.

(B) Test whether or not $\Phi_{k_1}(k_1) \le x$. In case $\neg\Phi_{k_1}(k_1) \le x$, output $\langle k_1,k_2,0\rangle$ and request the next input. Otherwise, distinguish the following cases.

(β1) If t_x^+ is a singleton set, then compute the minimal y such that $a^{\langle k_1,k_2\rangle}b^2 \notin L_{\langle k_1,k_2,y\rangle}$. Output $\langle k_1,k_2,y\rangle$ and request the next input.

(β2) Otherwise, compute $z = max\{n \,|\, t_x^+ \subseteq L_{\langle k_1,k_2,n\rangle}\}$. Output $\langle k_1,k_2,z\rangle$ and request the next input."

It remains to show that M infers L from t. Suppose $a^{\langle k_1,k_2\rangle}c^m \in t_x^+$ for some $m \in \mathbb{N}$. In accordance with the definition of $\mathcal{L}$ we may conclude that $\Phi_{k_1}(k_1)\downarrow$ as well as $\Phi_{k_2}(k_2)\downarrow$. Moreover, in this case L is already uniquely determined. Hence, M converges to a correct hypothesis.

Next we suppose that no string of the form $a^{\langle k_1,k_2\rangle}c^m$ does appear in t. By applying the same argumentation as in the proof of Theorem 4 it can be easily shown that M successfully handles this case, too. This proves the lemma.

Note that the IIM defined above performs, if ever, only one mind change not caused by an inconsistency. However, this mind change may violate the dual weak-monotonicity constraint. Anyway, in order to show $\mathcal{L} \notin WMON^d\!-\!TXT$ we need some additional insight into the behavior of *any* IIM inferring $\mathcal{L}$. The following lemma provides the desired information.

Lemma 2. *Let $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ be any class preserving hypothesis space for $\mathcal{L}$ and let M be any IIM witnessing $\mathcal{L} \in LIM\!-\!TXT$ with respect to $\mathcal{G}$. Then we have:*
For all k_2 there are numbers $k_1, x, j \in \mathbb{N}$ such that

(1) $M(t_x) = j$,

(2) $a^{\langle k_1,k_2\rangle}b^{x+2} \in L(G_j)$,

(3) $\Phi_{k_1}(k_1) > x+1$ *and* $\varphi_{k_1}(k_1)\downarrow$,

where t is the lexicographically ordered text of $L_{\langle k_1,k_2,0\rangle}$.

Proof. Suppose the converse. Then there is a k_2 such that for all k_1, x, j we have: $M(t_x) = j$ and $a^{\langle k_1,k_2\rangle}b^{x+2} \in L(G_j)$ implies $\Phi_{k_1}(k_1) \le x+1$ or $\Phi_{k_1}(k_1)\uparrow$.

Assuming the latter statement we have the following claim.

Claim. Provided the latter statement is true, any program for M may be used to obtain *non-effectively* an algorithm deciding "$\varphi_{k_1}(k_1)\downarrow$."

By assumption, there is a k_2 such that for all k_1, n, x: If (1) and (2) are fulfilled, then either $\Phi_{k_1}(k_1) \le x+1$ or $\Phi_{k_1}(k_1)\uparrow$. Using this k_2 we can define the following algorithm $\mathcal{A}$.

Algorithm $\mathcal{A}$: "On input k_1 execute (A1) and (A2).

(A1) Generate successively the lexicographically ordered text t of $L_{\langle k_1,k_2,0\rangle}$ and compute $M(t_0)$, $M(t_1)$, ... until the first hypothesis j satisfying $a^{\langle k_1,k_2\rangle}b^{x+2} \in L(G_j)$

is found.
Let x_0 be the least x such that $M(t_x) = j$ and $a^{\langle k_1, k_2\rangle} b^{x+2} \in L(G_j)$.

(A2) Test whether $\Phi_{k_1}(k_1) \leq x_0 + 1$.
In case it is, output "$\varphi_{k_1}(k_1)\downarrow$."
Otherwise, output "$\varphi_{k_1}(k_1)\uparrow$" and stop."

First we observe that M has to infers $L_{\langle k_1, k_2, 0\rangle}$ from its lexicographically ordered text. Hence, M should eventually output a hypothesis j such that $a^{\langle k_1, k_2\rangle} b^{x+2} \in L(G_j)$. Moreover, since membership in $L(G_j)$ is uniformly decidable for all $j \in \mathbb{N}$ and all $s \in \{a, b, c\}^*$, the described test can be performed effectively. Furthermore, Instruction (A2) can be accomplished effectively too. Hence, $\mathcal{A}$ is an algorithm and the execution of (A1) and (A2) must eventually terminate. Finally, by assumption we immediately obtain the correctness of $\mathcal{A}$'s output. This proves the claim.

On the other hand, the halting problem is algorithmically undecidable. Therefore, Lemma 2 follows.

***Lemma* 3**. $\mathcal{L} \notin WMON^d{-}TXT$

Proof. Suppose the converse, i.e., there are a class preserving hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ and an IIM M such that M $WMON^d{-}TXT$ identifies $\mathcal{L}$ with respect to $\mathcal{G}$. Moreover, without loss of generality we may assume that M exclusively produces consistent hypotheses (cf. Theorem 10).

Claim 1. Given $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ and any program for M witnessing $\mathcal{L} \in WMON^d{-}TXT$, one can *effectively* construct an algorithm deciding whether or not $\varphi_{k_2}(k_2)\downarrow$.

Let $K = \{k \mid \varphi_k(k)\downarrow\}$ and let $j_0, j_1, j_2, \ldots$ be any fixed effective enumeration of K. We define an algorithm $\mathcal{A}$ as follows.

***Algorithm* $\mathcal{A}$:** "On input k_2 execute (A1) and (A2).

(A1) For $k_1 = j_0,\ j_1,\ j_2, \ldots$ compute successively the lexicographically ordered texts $t^0,\ t^1,\ t^2, \ldots$ for $L_{\langle j_0, k_2, 0\rangle},\ L_{\langle j_1, k_2, 0\rangle}, \ldots$, respectively as well as by dovetailing

$M(t^0_0),\ M(t^0_1),\ M(t^0_2),\ \ldots,$
$M(t^1_0),\ M(t^1_1),\ M(t^1_2),\ \ldots,$
–
–
–
$M(t^r_0),\ M(t^r_1),\ M(t^r_2),\ \ldots,$
–
–
–

until the first hypothesis $j = j(j_z, k_2)$ is found that satisfies

(α1) $M(t^z_x) = j$,
(α2) $a^{\langle j_z, k_2\rangle} b^{x+2} \in L(G_j)$, and
(α3) $\Phi_{j_z}(j_z) > x + 1$.

(* By Lemma 2, the execution of (A1) has to terminate *)

(A2) Let $e =_{df} \langle j_z, k_2 \rangle$. Furthermore, we define $\hat{t}^z_{x+m}$ as follows:

$\hat{t}^z_{x+m} = \underbrace{a^e b, \ ..., \ a^e b^{x+1}}_{=t^z_x}, \ \underbrace{a^e b^{x+1}, \ ..., \ a^e b^{x+1}}_{m-\text{times}}$

For $m = 0, \ 1, \ 2, \ ...$ execute in parallel (β1) and (β2) until (β3) or (β4) happens.

(β1) Test whether $\Phi_{k_2}(k_2) = m$.

(β2) Compute $j_{x+m} = M(\hat{t}^z_{x+m})$.

(β3) $\Phi_{k_2}(k_2) = m$ is verified. Then output "$\varphi_{k_2}(k_2)\downarrow$."

(β4) In (β2) a hypothesis $j_{x+m} = M(\hat{t}^z_{x+m})$ is computed such that $a^e b^{x+2} \notin L(G_{j_{x+m}})$. Then output "$\varphi_{k_2}(k_2)\uparrow$" and stop."

It remains to show that $\mathcal{A}$ behaves correctly.

Claim 2. On every input k_2, the algorithm $\mathcal{A}$ terminates.

As we have already mentioned, by Lemma 2 we know that the execution of (A1) has to terminate. Hence, it suffices to show that either (β3) or (β4) happens. Suppose, (β3) does not happen. Then, for all $m \in \mathbb{N}$ we have $\neg\Phi_{k_2}(k_2) = m$. Consequently, $\Phi_{k_2}(k_2)\uparrow$. Therefore, $L_{\langle j_z, k_2, y\rangle} \subseteq \{a,b\}^+$ for all $y \in \mathbb{N}$ by definition of $\mathcal{L}$. Due to (α3) we know that $\Phi_{j_z}(j_z) > x+1$, and thus $\hat{t}^z =_{df} (\hat{t}^z_{x+m})_{m\in\mathbb{N}}$ is a text for the finite language $L = \{a^e b^n \mid 1 \le n \le x+1\}$ belonging to $\mathcal{L}$. Hence, M, when successively fed $\hat{t}^z$, has eventually to output a hypothesis j_{x+m} such that $a^e b^{x+2} \notin L(G_{j_{x+m}})$. Thus, ($\beta$4) must happen. This proves the claim.

Claim 3. Algorithm $\mathcal{A}$ works correctly.

Obviously, if (β3) happens then $\varphi_{k_2}(k_2)$ is indeed defined. Suppose, (β4) happens. We have to show that $\varphi_{k_2}(k_2)\uparrow$. Assume $\varphi_{k_2}(k_2)\downarrow$. Then $\Phi_{k_2}(k_2)\downarrow$, by the definition of a complexity measure.

Since (β4) happens, there is an $m \in \mathbb{N}$ such that M outputs the guess j_{x+m}, $a^e b^{x+2} \notin L(G_{j_{x+m}})$, when fed the initial segment $\hat{t}^z_{x+m}$ of a text for $L_{\langle j_z, k_2, 0\rangle}$. Let j_{x+r} $(0 \le r < m)$ denote M's last guess before outputting j_{x+m}. Since $a^e b^{x+2} \in L(G_{j_{x+r}})$ and $a^e b^{x+2} \notin L(G_{j_{x+m}})$, we may conclude that $L(G_{j_{x+r}}) \neq L(G_{j_{x+m}})$. Furthermore, M exclusively produces consistent hypotheses. Taking additionally into account that $range(\hat{t}^z_{x+r}) = range(\hat{t}^z_{x+m})$ it follows that the mind change from j_{x+r} to j_{x+m} cannot be caused by an inconsistency. Finally, by the definition of $\mathcal{L}$ we know that all languages that comprise at least $range(\hat{t}^z_{x+r})$ are incomparable because $\varphi_{k_2}(k_2)\downarrow$. Since $\mathcal{G}$ is a class preserving hypothesis space for $\mathcal{L}$, we obtain that the mind change from j_{x+r} to j_{x+m} violates the dual weak-monotonicity constraint, a contradiction.

This proves the claim. Thus the lemma is shown and the theorem follows. □

We next present a figure relating all types of class preserving monotonic language learning to all types of dual monotonic inference of indexed families. A directed edge from vertex A to vertex B indicates that A is a proper subset of B. Note that missing edges between learning types not connected by a path in the underlying graph are incomparable.

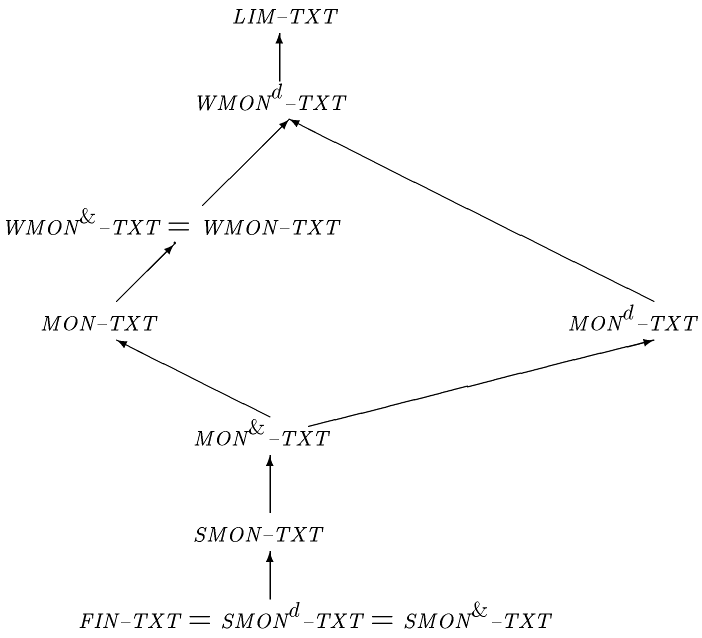

Figure 1

Finally, we want to deal with the combination of monotonic and consistent inference. As we shall see, the consistency demand does not limit the capabilities of monotonic and dual monotonic learners, respectively.

Theorem 10.

(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 other parts can be proved in a similar fashion. Obviously, $WMON\!-\!CONS\!-\!TXT \subseteq WMON\!-\!TXT$. For the converse, let $\mathcal{L}$ be any fixed indexed family of recursive languages, and let M be an IIM that weak-monotonically learns $\mathcal{L}$ with respect to some hypothesis space $(G_j)_{j\in\mathbb{N}}$. We define an IIM $\hat{M}$ as follows:

Let $L \in \mathcal{L}$, $t \in text(L)$, let $x \in \mathbb{N}$. When fed t_x, the machine $\hat{M}$ simulates M's computation on t_x. In case M does not produce any hypothesis, $\hat{M}$ requests the next input and outputs nothing. Otherwise, let j_z be M's last guess. The machine $\hat{M}$ tests whether $t_x^+ \subseteq L(G_{j_z})$. 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, it follows that $\hat{M}$ $WMON\!-\!CONS\!-\!TXT$–identifies L. We omit the details. $\square$

A closer look to the proof above shows that we mainly exploited the uniform recursiveness of all languages in the relevant hypothesis space as well the fact that the IIMs are not necessarily *responsive.* Following Angluin [1], an IIM M is said to responsively infer an indexed family $\mathcal{L}$, if M, when fed any initial segment of a text for any language in $\mathcal{L}$, is always required to output a hypothesis. Nevertheless, if one allows class comprising hypothesis spaces $\mathcal{G}$, that is, weakening the condition $\mathcal{L} = L(\mathcal{G})$ to $\mathcal{L} \subseteq L(\mathcal{G})$, then consistency and responsiveness may be combined with all types of monotonic inference without any limitation concerning the inference power. On the other hand, if one insists on class preserving hypothesis spaces, then IIMs that are consistent, responsive and monotonic are almost always slightly weaker than monotonic ones (cf. Example 4). For all models of unconstrained learning, the combination of consistency and responsiveness does not constitute a restriction (cf. Proposition 1 and Angluin [1]). These results impressively show that the choice of the hypothesis space may not only influence the inferring power at all (cf. Lange and Zeugmann [34]) but may or may not allow the combination of postulates of naturalness.

Our next example shows that responsiveness in combination with consistency is a severe requirement for class preserving strong-monotonic, monotonic and weak-monotonic learning, respectively.

Example 4. Consider the following indexed family $\mathcal{L} = (L_{\langle k,j\rangle})_{k,j\in\mathbb{N}}$. For all $k \in \mathbb{N}$, let $L_{\langle k,0\rangle} = \{a^k b^m \mid m \in \mathbb{N}^+\}$. For $j > 0$ we distinguish the following cases.

Case 1. $\neg\ \Phi_k(k) \leq j$

Then we set $L_{\langle k,j\rangle} = L_{\langle k,0\rangle}$.

Case 2. $\Phi_k(k) \leq j$

We set $L_{\langle k,j\rangle} = \{a^k b\}$.

On the one hand, it is not hard to verify that $\mathcal{L} \in ESMON{-}TXT$. On the other hand, suppose any consistent as well as responsive IIM M which $WMON-TXT$–identifies $\mathcal{L}$ with respect to a class preserving hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$. Then, M may be used to decide, for every $k \in \mathbb{N}$, whether or not $\varphi_k(k)$ is defined. In doing so, it suffices to simulate M when fed the string $a^k b$. Since M is a consistent and responsive IIM, M has to output a hypothesis j such that $a^k b \in L(G_j)$. Finally, since $\mathcal{G}$ is a class preserving space of hypotheses, it can be easily verified whether or not $L(G_j) = \{a^k b\}$. In case it is, $\varphi_k(k)$ has to be defined. Otherwise, it follows $L(G_j) = L_{\langle k,0\rangle}$. Therefore, $\varphi_k(k)$ has to be undefined, since otherwise M has produced an overgeneralized hypothesis when fed an initial segment of a text for $L = \{a^k b\}$. Clearly, this contradicts our assumption that M $WMON-TXT$–identifies every language $L \in \mathcal{L}$. $\diamond$

4. Monotonic inference from informant

This section is devoted to give an overall picture relating all models of class preserving monotonic and dual monotonic language learning from both text and informant to one another. Let us begin with a quite simple example which illustrates the differences between learning from positive data and learning from both positive and negative data.

Example 5. We set $L_0 = \{a\}^+$, $L_k = L_0 \setminus \{a^k\}$, for $k \geq 1$, and define $\mathcal{L}_{inf} = (L_k)_{k\in\mathbb{N}}$. An IIM M which infers $\mathcal{L}_{inf}$ from informant may work as follows: Initially, it outputs the guess 0. As long as some $(a^k, -)$ does not appear in the informant, the machine M outputs 0. In case it does, M performs a mind change and guesses k. This hypothesis is then repeated in every subsequent step. Clearly, the machine M learns $\mathcal{L}$. Moreover, M witnesses $\mathcal{L} \in SMON^d{-}INF \cap MON{-}INF$. As we shall see in the demonstration of Theorem 16, $\mathcal{L}_{inf} \notin LIM{-}TXT$. $\diamond$

Our first result shows that, in contrast to the text case, conservative IIMs are exactly as powerful as unconstrained machines, if learning from informant is investigated.

Proposition 2. $LIM{-}INF = CONSERVATIVE{-}INF$

Proof. In order to show $LIM - INF \subseteq CONSERVATIVE{-}INF$ one simply has to recognize that any indexed family $\mathcal{L}$ can be identified by an IIM realizing the *identification by enumeration principle* (cf. Gold [16]). Such a machine performs exclusively justified mind changes. Hence, it is conservative. $\square$

Proposition 2 allows the following corollary which actually shows that the inferring power of weak-monotonic and dual weak-monotonic IIMs coincides with the capabilities of unconstrainted ones. To see this recall that by definition every conservative IIM simultaneously fulfills the weak-monotonicity as well as the dual weak-monotonicity constraint.

Corollary 11. $WMON{-}INF = WMON^d{-}INF = WMON^{\&}{-}INF = LIM{-}INF$

The next theorem in this section establishes a hierarchy between the different types of monotonicity.

Theorem 12.

(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 $MON{-}TXT \setminus SMON{-}INF \neq \emptyset$. Since $MON{-}TXT \subseteq MON{-}INF$, we can conclude that $SMON{-}INF \subset MON{-}INF$. For the demonstration that $WMON{-}INF \setminus MON{-}INF \neq \emptyset$, the reader is referred to the proof of Theorem 16, Claim B and C, where we prove $WMON{-}TXT \setminus MON{-}INF \neq \emptyset$. Finally, Assertions (2), (3) and (4) have been established in Lange and Zeugmann [37]. $\square$

The proper inclusions among the classes above remain valid even if one restricts oneself to consider only families of regular languages. The next theorem shows the lower bounds of these hierarchies.

Theorem 13.

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

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

Proof. The part $FIN{-}INF \subseteq SMON{-}INF$ is again obvious. The proper set inclusion is obtained by observing that $\mathcal{L}_{finvar} \notin FIN{-}INF$, where $\mathcal{L}_{finvar} = \{L \cup L_{fin} \,|\, L_{fin} \in \mathcal{L}_{fin}\}$, that is, the family defined in Example 1. On the other hand, $\mathcal{L}_{finvar} \in SMON{-}TXT$ as we have shown above (cf. Example 1). Hence, Assertion (1) follows. Furthermore, by definition $FIN{-}INF \subseteq SMON^d{-}INF$. The remaining part of Assertion (2) is an immediate consequence of Theorem 12, Part (4). □

Summarizing the results above, one can easily derive the following corollary:

Corollary 14.

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

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

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

Figure 2 gives an overview of the relationships among the various types of monotonic and dual monotonic inference from informant. The semantics of Figure 2 is analogous to that of Figure 1.

Monotonic versus Dual Monotonic Inference from Informant

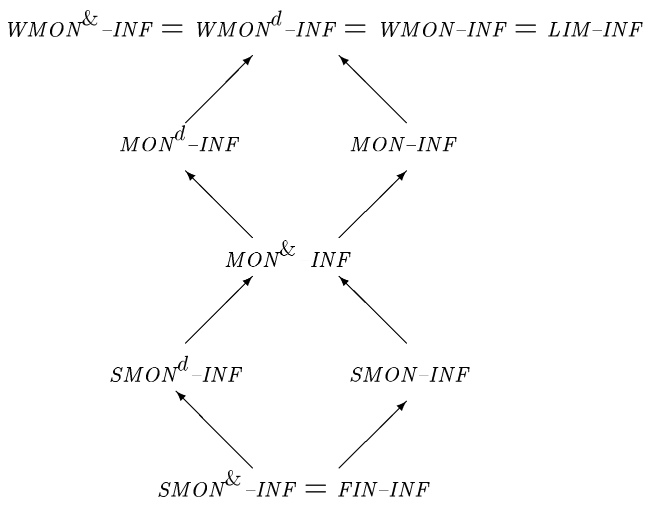

Figure 2

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

Theorem 15.

(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 by methods analogous to those used to establish Theorem 10. □

We next 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 [16] has shown that the inference from positive data alone is much weaker than inference on informant. Consequently, one would expect monotonic inference on text to be less powerful than monotonic learning from informant. However, the more interesting question is whether one can strengthen the monotonicity requirement in case one can use both positive and negative data rather than just positive data. From another perspective, this amounts to asking whether the weakening of the notion of monotonicity allows the infer indexed families from text which can only be inferred from informant in the stronger monotonic sense. Our next theorem shows that the various classes are actually almost always incomparable, that is, in general, one cannot trade monotonicity versus information presentation.

Theorem 16.

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

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

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

(4) $FIN\!-\!INF \;\#\; MON\!-\!TXT$

Proof. The first part we prove here is $MON\!-\!INF \setminus LIM\!-\!TXT \neq \emptyset$. As we have already seen, the family $\mathcal{L}_{inf}$, that is, the family introduced in Example 5 belongs to $MON\!-\!INF$. Thus, Claim A below completes the desired separation.

Claim A. $\mathcal{L}_{inf} \notin LIM\!-\!TXT$

By Proposition 1, $LIM\!-\!TXT = ELIM\!-\!TXT$. Now, due to Theorem 1 of Angluin [1], it suffices to show that there is no finite tell-tale for $L_0 = \{a\}^+$. Suppose the converse, that is, there is a recursively enumerable finite set T_0 satisfying:

(α) $T_0 \subset L_0$, and

(β) for all $j > 1$, if $T_0 \subseteq L_j$, then $L_j \not\subset L_0$.

Let $z = max\{|s| \mid s \in T_0\}$. Then, in accordance with the definition of the languages L_k, we immediately get $L_0 \setminus \{a^{z+1}\} = L_{z+1} \subseteq T_0$. Moreover, $L_{z+1} \subset L_0$ in contradiction to condition (β). This shows that $MON\!-\!INF \setminus LIM\!-\!TXT \neq \emptyset$.

Next we show $WMON-TXT \setminus MON-INF \neq \emptyset$. Let $L_0 = \{a\}^+$. For all $k,\ j \in \mathbb{N}$ with $m > k \geq 1$, let $L_k = \{a^z \mid 1 \leq z \leq k\} \cup \{b^k\}$ and $L_{k,m} = L_k \cup \{a^m\}$. Finally, we denote by $\mathcal{L}_{wmon}$ a canonical enumeration of all languages: L_k, $k \in \mathbb{N}$, and $L_{k,m}$, $k, m \in \mathbb{N}$ and $m > k$.

Claim B. $\mathcal{L}_{wmon} \in WMON-TXT$

The desired 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 language $\{a\}^+$. In case a string of the form b^k appears in t the machine M outputs a grammar for L_k. This guess remains unchanged until there appears a string a^m in the text t with $m > k$. Then M outputs a grammar for $L_{k,m}$.

Looking at the definition of M we immediately observe that M changes its mind only in case its current guess is inconsistent with the new data. We can easily argue that M weak-monotonically learns $\mathcal{L}_{wmon}$.

Finally, we have to prove the following claim:

Claim C. $\mathcal{L}_{wmon} \notin MON-INF$

Suppose the converse, i.e., assume there is an IIM M and a hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ such that M monotonically learns $\mathcal{L}_{wmon}$ from informant with respect to $\mathcal{G}$. Let i be an informant for $\{a\}^+$. Since $L_0 \in \mathcal{L}_{wmon}$ there must be an x such that $j_x = M(i_x)$ and $L(G_{j_x}) = L_0$. Let $k = max\{|s| \mid s \in (i_x^+ \cup i_x^-)\} + 1$. Next, we consider an informant $\hat{\imath}$ for the language L_k that contains i_x as prefix. Hence, there must be a number z such that M on $\hat{\imath}_{x+z}$ outputs a grammar j_{x+z} for L_k. But now we can extend $\hat{\imath}_{x+z}$ in a canonical manner to an informant i_{fool} for $L_{k,m}$ where $m = max\{|s| \mid s \in (i_{x+z}^+ \cup i_{x+z}^-)\} + 1$. It is easy to see that M either does not behave monotonically on i_{fool} or it does not infer $L_{k,m}$. This proves Claim C, 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. It remains to show that $FIN-INF \setminus MON-TXT \neq \emptyset$. This is done via the family $\mathcal{L}_{fool}$ defined as follows: For each $k \in \mathbb{N}$, let $L_k = \{a\}^+ \setminus \{a^{k+1}\}$. We define $\mathcal{L}_{fool}$ to be the family of all languages L_k. Obviously, $\mathcal{L}_{fool}$ is an indexed family. First, we show that $\mathcal{L}_{fool} \notin MON-TXT$. Suppose it is, i.e., there is an IIM M and a hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ such that M monotonically identifies $\mathcal{L}_{fool}$ from text with respect to $\mathcal{G}$. 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 $L(G_{j_{z-1}}) = L_0$. In case it does not, we are done, since M does not identify L_0 from its lexicographically ordered text. Then, we input $a, a^{z+2}, a^{z+3}, ...$ until M produces a correct guess j_r for L_z. Again, there has to be an $r \geq z + 2$ such that M on $a^2, a^3, \ldots, a^z, a, a^{z+2}, a^{z+3}, \ldots, a^r$ outputs j_r because otherwise M fails to infer L_z. Now we proceed as follows. We set $t_{fool} = a^2, a^3, ..., a^z, a, a^{z+2}, ..., a^r, a^{z+1}, a^{r+2}, a^{r+3}, ...$, that is, t_{fool} is a text for L_r. Since $a^{z+1} \in L_0 \cap L_r$ but $a^{z+1} \notin L_z \cap L_r$, we get $L_0 \cap L_r \not\subseteq L_z \cap L_r$. Thus, M violates the monotonicity constraint when fed t_{fool}, a contradiction.

It remains to show that $\mathcal{L}_{fool} \in FIN-INF$. The IIM M finitely inferring $\mathcal{L}_{fool}$ from informant works as follows: Let $L \in \mathcal{L}_{fool}$, let $i \in info(L)$, and let $x \in \mathbb{N}$. 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-1} and stops. □

Theorem 16 has the following direct consequence.

Corollary 17.

(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 previous theorem can be sharpened to $SMON-INF \setminus LIM-TXT \neq \emptyset$. The answer to this question puts the complexity of strong-monotonic inference in the correct perspective.

Theorem 18. $SMON\!-\!INF \subset WMON\!-\!TXT$

Proof. It suffices to show $SMON\!-\!INF \subseteq WMON\!-\!TXT$ because then, by Theorem 16, Assertion (1), one directly obtains $WMON\!-\!TXT \setminus SMON\!-\!INF \neq \emptyset$. Let M be an IIM that infers $\mathcal{L}$ strong-monotonically from informant. Without loss of generality, we may assume M is total, that is, defined on any finite sequence of strings from $\Sigma^* \times \{+,-\}$, that is, M either outputs a hypothesis or it request the next input. The following argument verifies that this reasonable:

Suppose $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ is a hypothesis space and $\hat{M}$ is an IIM that strong-monotonically infers the family $\mathcal{L}$ from informant with respect to $\mathcal{G}$. We construct an IIM M with the desired property that uses $\mathcal{G}$ as hypothesis space, too. Let $i_x = (s_0, b_0), ..., (s_x, b_x)$ be any finite sequence. Then the IIM M first checks whether 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 an L_k does not exist, M does not output a guess and goes on to request the next input. Otherwise, M on input i_x outputs exactly what $\hat{M}$ outputs. Note that in the second case $\hat{M}$ has to be defined because the input is an initial segment of some informant for L_k. Moreover, M infers $\mathcal{L}$ strong-monotonically. Let $L \in \mathcal{L}$, $y = min\{z \mid L_z = L\}$, and let i be any informant of L. Consequently, on input i_x, for all $x \geq y$, M finds at least one language L_k such that $k \leq x$, $i_x^+ \subseteq L_k$, and $i_x^- \subseteq co-L_k$, that is, 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 skips some hypotheses $\hat{M}$ might have computed. Since set inclusion is a transitive relation, M works strong-monotonically. This completes the verification.

We now construct an IIM M' weak-monotonically identifying $\mathcal{L}$ from text with respect to $\mathcal{G}$. This is done as follows: Let $L \in \mathcal{L}$ and let $t = (s_j)_{j\in\mathbb{N}}$ be an arbitrarily fixed text for L. Then the desired machine M' is defined as follows:

IIM *M'*: "On input t_x do the following: Rearrange the strings contained in t_x in lexicographical order without repetitions. Let $t'_k = w_0, ..., w_k$, $k \leq x$ the sequence obtained.
Compute $A_x := \{(w_0,+), ..., (w_k,+)\} \cup \{(v,-) \mid v \in \Sigma^* \setminus t_x^+,\ |v| \leq max\{x,\ |w_k|\}\}$
Determine $i(x)_{z(x)}$ by rearranging the elements of A_x in lexicographical order with

respect to the first component.
(* Note that $i(x)_{z(x)}$ is not necessarily an initial segment of an informant of L, since it may contain some pair $(v, -)$ such that $v \in L$. *)
Compute successively $M(i(x)_0)$, $M(i(x)_1)$, ..., $M(i(x)_{z(x)})$ until M outputs the first hypothesis j_r satisfying $i(x)^+_{z(x)} \subseteq L(G_{j_r})$ and $i(x)^-_r \subseteq co-L(G_{j_r})$.
(*Observe that, by construction, every string $(w, +) \in i(x)^+_{z(x)}$ indeed belongs to L. *)
In case j_r has been found, output it. Otherwise, output nothing, and request s_{x+1}."

It remains to show that M' weak-monotonically learns $\mathcal{L}$ from text with respect to $\mathcal{G}$. Let $L \in \mathcal{L}$, and $t \in text(L)$.

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

Let i^{ord} be the lexicographically ordered informant for L. By assumption, M strong-monotonically infers L from i^{ord} with respect to $\mathcal{G}$. Hence, there exists a smallest $m \in \mathbb{N}$ such that $M(i^{ord}_m) = j_m$ and $L(G_{j_m}) = L$. Moreover, since t is a text for L there is a smallest index x_0 satisfying $i^{ord,+}_m \subseteq t^+_x$ for all $x \geq x_0$. Now, let us have a closer look to the initial segments $i(x)_{z(x)}$ of conjectured informants for L obtained from t_x as described above. By construction we immediately obtain $t^+_x = i(x)^+_{z(x)} \subseteq i(x+1)^+_{z(x+1)}$ for all $x \in \mathbb{N}$. Thus, $i^{ord,+}_m \subseteq i(x)^+_{z(x)}$ for every $x \geq x_0$. Moreover, $i^{ord,-}_m \subseteq i(x)^-_{z(x)}$ for all $x \geq x_0$, since M' classifies every string $s \notin t^+_x$ of length less than or equal to $y = max\{x,\ max\{|s| \mid s \in t^+_x\}\}$ as $(s, -)$. Finally, $i(x_0)_m = i(x)_m$ for all $x \geq x_0$. Since $L(G_{j_m}) = L$, therefore we have:

$$i(x)^+_{z(x)} \subseteq L(G_{j_m}) \text{ and } i(x)^-_m \subseteq co - L(G_{j_m}) \text{ for every } x \geq x_0 \tag{1}$$

Hence, it suffices to argue that $M'(t_x) = j_m$ for almost all $x \geq x_0$. Obviously, if m is the least index r satisfying $i(x)^+_{z(x)} \subseteq L(G_{j_r})$ and $i(x)^-_r \subseteq co-L(G_{j_r})$ we are done. Suppose the converse, i.e., there exists an $r < m$ such that $i(x)^+_{z(x)} \subseteq L(G_{j_r})$ and $i(x)^-_r \subseteq co - L(G_{j_r})$ for infinitely many $x \geq x_0$. Now, $i(x)_r$ is an initial segment of $i(x_0)_m$ and, therefore, of i^{ord}, too, since $x \geq x_0$ and $r < m$. Hence, $L(G_{j_r}) \subset L$, since M is strong-monotonic and m is the least index n such that $L = L_{M(i^{ord}_n)}$. Consequently, there exists a string $s \in L \setminus L(G_{j_r})$. Since $t \in text(L)$, there must be an $\hat{x}$ such that $s \in t^+_x$ for all $x \geq \hat{x}$. Therefore, $i(x)^+_{z(x)} \not\subseteq L(G_{j_r})$ for all $x \geq \hat{x}$, a contradiction.

Claim 2. M' is weak-monotonic.

It remains to show that every mind change of M' is forced by an inconsistency with the text t. Let $j_x = M'(t_x)$ and $j_{x+1} = M'(t_{x+1})$ such that $j_x \neq j_{x+1}$. Note that $M'(t_x)$ denotes the last guess output by M' when successively fed t_x (cf. Section 2). Hence, for each mind change there is an x such that $M'(t_x) \neq M'(t_{x+1})$. Furthermore, by construction there exists r_1 and r_2 such that $j_x = j_{r_1} = M(i(x)_{r_1})$ and $j_{x+1} = j_{r_2} = M(i(x+1)_{r_2})$. We distinguish the following cases.

Case 1. $i(x)_{r_1} = i(x+1)_{r_1}$

Hence, $i(x)^-_{r_1} = i(x+1)^-_{r_1}$. Since j_{r_1} is rejected, M' must have verified $i(x+1)^+_{z(x+1)} \not\subseteq L(G_{j_{r_1}})$. But this directly implies $t^+_{x+1} \not\subseteq L(G_{j_{r_1}})$.

Case 2. $i(x)_{r_1} \neq i(x+1)_{r_1}$

By construction, we may directly conclude $i(x)^-_{r_1} \neq i(x+1)^-_{r_1}$. Therefore, we additionally get $i(x+1)^-_{r_1} \subset i(x)^-_{r_1}$. But this directly implies that there is a string $s \in t^+_{x+1}$ such that $s \notin L(G_{j_{r_1}})$. Hence, whatever r_2 is, the mind change has been caused by inconsistency with the text t. $\square$

The latter theorem directly allows the following corollary.

Corollary 19. $FIN-INF \subset WMON-TXT$

As regards dual strong-monotonic inference from 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 $\mathcal{L}_{inf}$ introduced in Example 5 where we have shown that $\mathcal{L}_{inf}$ belongs to $SMON^d{-}INF$. On the other hand, we know already $\mathcal{L} \notin LIM{-}TXT$. This completes the proof. $\square$

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 can 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 remaining to be proved is $MON^d{-}TXT \setminus SMON^d{-}INF \neq \emptyset$. Again, we use the indexed family $\mathcal{L}_{wmon}$ introduced in the proof of Theorem 16. It suffices to show that the IIM M defined there to witness $\mathcal{L}_{wmon} \in WMON\text{–}TXT$ works as well dual monotonically. To see this recall that M performs at most two mind changes, that is, it eventually guesses subsequently the languages $L_0 = \{a\}^+$, $L_k = \{a^z \mid 1 \leq z \leq k\} \cup \{b^k\}$ and finally $L_{k,m} = L_k \cup \{a^m\}$. Then we have $co{-}L_0 \cap co{-}L_{k,m} \subseteq co{-}L_k \cap co{-}L_{k,m} \subseteq co{-}L_{k,m}$, since $co - L_{k,m} = co - L_k \setminus \{a^m\}$. Hence $co - L_k \cap co - L_{k,m} = co - L_{k,m}$, that is, M does work dual monotonically.

$\mathcal{L}_{wmon} \notin MON{-}INF$ have been shown in Theorem 16, Claim C. This completes the proof. $\square$

With the following figure we summarize the known results concerning monotonic language learning (cf. Lange and Zeugmann [31, 32]). Each learning type is represented as a vertex in a directed graph. A directed edge from vertex A to vertex B indicates that A is a proper subset of B, a bidirectional edge represents $A = B$, and no edge between vertices not connected by a directed path implies that A and B are incomparable.

Monotonic Learning from Text versus Monotonic Learning from Informant

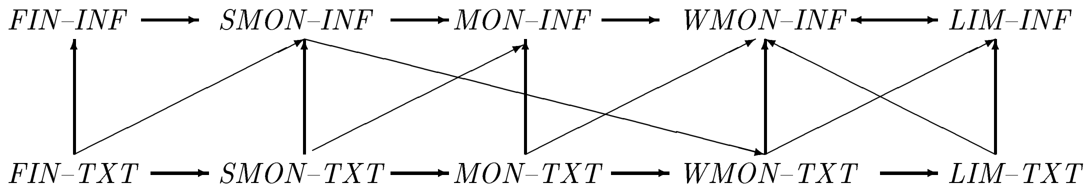

Figure 3

The next figure shows the relations between the defined models of class preserving dual monotonic inference (cf. Lange and Zeugmann [37]). The semantics of Figure 4 is analogous to that of Figure 3. On comparing with Figure 3, the similarities as well as the differences between the various types of monotonic and dual monotonic inference are clearly illustrated.

Dual Monotonic Learning from Text versus Dual Monotonic Inference from Informant

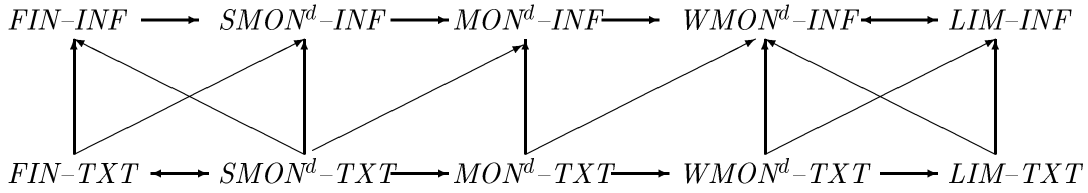

Figure 4

After having clarified the principal relationships between the different notions of monotonicity, we next deal with the question what 'natural' families of languages can be inferred monotonically. This is the issue discussed 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 learned strong-monotonically or monotonically from text or informant, respectively. Pattern languages have been introduced by Angluin [2], thereby proving that the whole class of pattern languages can be inferred in the limit from positive data. Subsequently, Shinohara [45] dealt with polynomial time learnability of subclasses of pattern languages. Nix [41] outlined interesting applications of pattern inference algorithms. Recently, Kearns and Pitt [26] as well as Ko et al. [27] studied intensively pattern inference from positive and negative data in the PAC–learning model. Recently, Wiehagen and Zeugmann [51] studied the pattern languages with respect to their polynomial time learnability by consistent and inconsistent IIMs, respectively.

So let us define what are a pattern and a pattern language. Let $\Sigma = \{a, b, ...\}$ be any non-empty finite alphabet containing at least two letters, and let $X = \{x_0, x_1, ...\}$ be an infinite set of variables such that $\Sigma \cap X = \emptyset$. *Patterns* are non-empty strings over $\Sigma \cup X$, 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. From the viewpoint of applications is often desirable to learn the pattern languages with respect to a hypothesis space that exclusively consists of patterns. Moreover, it is highly desirable to choose the hypothesis space as small as possible. For that purpose we use the canonical form of patterns (cf. Angluin [2]). A pattern p is in *canonical form* provided that if k is the number of variables in p, then the variables occurring in p are precisely $x_0, ..., x_{k-1}$. Moreover, for every j with $0 \leq j < k-1$, the leftmost occurrence of x_j in p is left to the leftmost occurrence of x_{j+1} in p. If a pattern p is in canonical form then we refer to p as a canonical pattern. Let *Patc* denote the set of all canonical patterns. Clearly, for every pattern p there exists a unique $q \in Patc$ such that $L(p) = L(q)$. Finally, choose any repetition free effective enumeration $p_0, p_1, ...$ of *Patc* and define $PAT = (L(p_j))_{j\in\mathbb{N}}$. Then PAT establishes an indexed family (cf. Angluin [2]).

Using the techniques developed in Lange and Wiehagen [30], one can obtain the following result:

Theorem 23.

(1) $PAT \in FIN\!-\!INF$

(2) $PAT \notin FIN\!-\!TXT$

Proof. Suppose $L \in PAT$ 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 unique $k \in \mathbb{N}$ 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 such a k, the machine M starts the procedure described in Lange and Wiehagen [30] 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 a text for any pattern language L is also a text for $L(x_1)$. Hence, $L(x_1)$ cannot be finitely inferred from text. We omit the details. □

On the other hand, $PAT \in EWMON{-}TXT$ (cf. Angluin [1]). Hence, it is only natural to ask whether or not this result can be strengthened. Our answer is twofold. First, as long as one insists to learn the pattern languages with respect to any class preserving hypothesis space, the answer is negative, i.e., $PAT \notin MON{-}TXT$ (cf. Zeugmann, Lange and Kapur [53] for a detailed discussion). As a matter of fact, the latter result is in part based on the nondecidability of inclusion of pattern languages (cf. Jiang et al. [24]). Therefore, one might conjecture that for two potential hypotheses G and G', the decidability of '$L(G) \subseteq L(G')$' may help in the monotonic learnability of a family. Even for learning on *informant*, this fails to be an adequate guarantee.

Theorem 24. *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 Example 2. All these languages are regular. However, in a manner analogous to the proof of Claim C in Theorem 16, one can show that there cannot be an IIM M inferring all the regular languages monotonically from informant. □

Second, having the negative result $PAT \notin MON{-}TXT$ in mind, it is reasonable to ask whether an appropriate choice of the hypothesis space may help to overcome the difficulty in designing an IIM that infers PAT monotonically. This is indeed the case, and we can do even better. We define the wanted hypothesis space as follows. Let $DPAT$ be the family of all languages L for which there are finitely many patterns $p_{i_1}, ..., p_{i_k} \in Patc$ such that $L = \bigcap_{j=1}^{k} L(p_{i_j})$. Clearly, $PAT \subset DPAT$. Our next theorem, which is based on an idea communicated to us by H. R. Beick [8], shows the way to a positive answer. But we prove even a slightly stronger result, namely the strong-monotonic learnability of the *whole* hypothesis space $DPAT$.

Theorem 25. $DPAT \in SMON{-}TXT$

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. 18), that is, given two context-sensitive grammars G_1 and G_2, one can effectively construct another context-sensitive 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 $(G_j)_{j \in \mathbb{N}}$ of grammars such that $DPAT = \{L(G_j) \mid j \in \mathbb{N}\}$.

We define an IIM M inferring $DPAT$ strong-monotonically from text as follows: Let $L \in DPAT$, let $t \in text(L)$, and $x \in \mathbb{N}$.

IIM *M*: "On input t_x do the following: 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 M strong-monotonically learns $DPAT$. Let $L \in DPAT$, that is, there are patterns $q_1, ..., q_\ell$ such that $L = \bigcap_{j=1}^{\ell} L(q_j)$. Let $m = min\{|s| \mid s \in L\}$.

Consequently, $|q_j| \leq m$ for all $j = 1, ..., \ell$. 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 $j = 1, ..., \ell$. Thus $\{1, ..., \ell\} \subseteq I_x$ for every $x \in \mathbb{N}$. Hence, every $j_x = M(t_x)$ satisfies $L(G_{j_x}) \subseteq L$. Furthermore, since $I_{x+k} \subseteq I_x$, for every $k \in \mathbb{N}$, we additionally get $L(G_{j_x}) \subseteq L(G_{j_{x+k}}) \subseteq L$. Consequently, M is a strong-monotonic IIM. Since $I_0 \setminus \{1, ..., \ell\}$ is finite, M must converge. Let j_{fin} be the limit of the sequence of hypotheses. Suppose $L(G_{j_{fin}}) \subset L$. Then there is at least one string $s \in L \setminus L(G_{j_{fin}})$. At the same time, since t is a text for L, there must be an $x \in \mathbb{N}$ such that $s \in t_x^+$. By 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 $s \in L(G_{j_{fin}})$. Consequently, M must converge to a correct grammar. □

The following corollary is immediate.

Corollary 26. $PAT \in CSMON{-}TXT$

This result witnesses that having the freedom to select a hypothesis space which comprises the target indexed family may increase the power of strong-monotonic IIMs. In the next section we study the power of class comprising learning under various monotonicity constraints in some more detail.

6. Class comprising monotonic learning

Within Section 3 and 4 we have related all models of class preserving monotonic and dual monotonic language learning to one another. Naturally, the question arises whether the overall picture remains unchanged, if one studies exact and class comprising learning, respectively. Under the requirement that a target indexed family $\mathcal{L}$ has to be learned with respect to the hypothesis space $\mathcal{L}$ itself all the separations previously obtained *remain valid*, no matter whether we consider language learning from text or learning from informant. On the other hand, in proving some of the stated inclusions (cf. Theorem 1 and Theorem 6) we heavily exploited the assumption that every language belonging to the chosen hypothesis space has to be learned itself. Obviously, this assumption does not longer hold, if class comprising hypothesis spaces are admissible. Therefore, one may expect that at least some of the established relations might change. And indeed, class comprising monotonic and dual monotonic learning from text has its own peculiarities.

Subsequently, we summarize the differences between class preserving and class comprising monotonic as well as dual monotonic learning from text. We start with the following result which actually shows that the dual weak-monotonicity constraint does not longer restrict the learning capabilities.

Theorem 27. $CWMON^d{-}TXT = ELIM{-}TXT$

Proof. Let $\mathcal{L}$ be any indexed family such that $\mathcal{L} \in CWMON^d{-}TXT$. By definition of dual weak-monotonic learning one directly obtains that $\mathcal{L} \in CLIM{-}TXT$. Taking Proposition 1 into account $\mathcal{L} \in ELIM{-}TXT$ follows.

Now, let $\mathcal{L} \in ELIM{-}TXT$. Hence, there is an IIM $\hat{M}$ that $ELIM{-}TXT$–identifies $\mathcal{L}$. The wanted hypothesis space $\mathcal{G}$ as well as the desired IIM M witnessing $\mathcal{L} \in CWMON^d{-}TXT$ are defined as follows. Let $\mathcal{G}$ be any fixed canonical enumeration of grammars generating all finite intersections of languages from $\mathcal{L}$. Consequently, $range(\mathcal{L}) \subseteq \{L(G_j) \mid j \in$

$\mathbb{N}\}$. Furthermore, let $L \in \mathcal{L}$, let $t \in text(L)$, and let $x \in \mathbb{N}$. We define:

IIM *M*: "On input t_x do the following: If $x = 0$ or $x > 0$ and M when successively fed t_{x-1} does not produce any guess, then goto (A). Else goto (B).

(A) Simulate $\hat{M}$ on input t_x. If $\hat{M}$, when successively fed t_x, outputs no guess then output nothing, and request the next input.
Otherwise, let $j = \hat{M}(t_x)$. Set $y = x$, output the canonical index for L_j in $\mathcal{G}$, and request the next input.

(B) Let $k = M(t_{x-1})$. Check whether or not $t_x^+ \subseteq L(G_k)$.
In case it does not, output the canonical index for $L_{\hat{M}(t_x)}$ in $\mathcal{G}$, set $y = x$, and request the next input.
In case it is, test whether or not $\hat{M}(t_y) = \hat{M}(t_x)$. If the latter test is fulfilled, output k, and request the next input.
Otherwise, output the canonical index for $L(G_k) \cap L_{\hat{M}(t_x)}$ in $\mathcal{G}$, and request the next input."

It remains to prove that M dual weak-monotonically infers $\mathcal{L}$ with respect to $\mathcal{G}$. Let $L \in \mathcal{L}$ and let $t \in text(L)$.

Claim 1. M fulfills the dual weak-monotonicity constraint.

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 $L(G_k) \cap L_{\hat{M}(t_x)}$, 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. This can be seen as follows. Let m and n be the least numbers such that $\hat{M}(t_m) = \hat{M}(t_{m+r})$ and $M(t_n) = M(t_{n+r})$ for all $r \in \mathbb{N}^+$. Suppose, $n < m$. Then we distinguish the following cases.

Case 1. $t_{n+r}^+ \subseteq L(G_{M(t_n)})$ for all $r \in \mathbb{N}^+$.

Since $m > n$ the IIM $\hat{M}$ performs at least one more mind change after having fed t_n. Hence, M has to change its mind from $M(t_n)$ to the canonical index for $L(G_{M(t_n)}) \cap L_{\hat{M}(t_{n+y})}$ for some $y > 0$, a contradiction.

Case 2. $t_{n+r}^+ \not\subseteq L(G_{M(t_n)})$ for some $r \in \mathbb{N}^+$.

Let r_0 be the least such r, i.e., $t_{n+r_0}^+ \not\subseteq L(G_{M(t_n)})$. Hence, M on input t_{n+r_0} does not output $M(t_{n+r_0-1})$, but instead it outputs the canonical index j of $L_{\hat{M}(t_{n+r_0})}$. If $M(t_{n+r_0-1}) \neq j$ we immediately obtain a contradiction. On the other hand, if $M(t_{n+r_0-1}) = j$ the IIM M has to detect an inconsistency on input t_{n+r_0+1}, too. Moreover, $t_{n+r_0}^+ \not\subseteq L_{\hat{M}(t_{n+r_0})}$. Consequently, $\hat{M}$ has to perform at least one more mind change in order to converge to a correct index k of L in $\mathcal{L}$. Hence, $L_{\hat{M}(t_{n+r_0})} \neq L_k$. By construction, M simulates that mind change. This contradicts our assumption that $M(t_n) = M(t_{n+r})$ for all $r \in \mathbb{N}^+$.

Finally, assume that $\hat{M}$ converges to k. Then it immediately follows that $L = L_k$. Hence, z, the index M converges to, cannot be the canonical index of L_k in $\mathcal{G}$. Thus, in accordance with its definition M converges to the canonical index for some $L(G_j) \cap L_k$. This contradicts $L \subset L(G_z)$. This proves the claim.

Claim 3. M converges on t.

By assumption, $\hat{M}$ converges on t, say to k. Hence, there are two cases to distinguish. Either M sometimes outputs the canonical index of L_k in $\mathcal{G}$. Hence, afterwards it cannot detect any inconsistency nor can it detect any mind change performed by $\hat{M}$. Thus, it converges to the canonical index of L_k in $\mathcal{G}$. In the second case, it sometimes outputs the canonical index for some $L(G_j) \cap L_k$. Since $\hat{M}$ converges to k, M cannot detect an additional mind change performed by $\hat{M}$. Hence, any further mind change of M has to be caused by a detected inconsistency. But if this happens, then M outputs the canonical index of L_k in $\mathcal{G}$ and we are back to Case 1. Hence, M converges. □

Clearly, the space of hypotheses constructed above will not necessarily meet the class-preserving requirement. Nevertheless, the same idea directly yields the following corollary.

Corollary 28. *If $\mathcal{L} \in CLIM-TXT$, then there is an indexed family $\hat{\mathcal{L}}$ at least containing all the languages in $\mathcal{L}$ such that $\hat{\mathcal{L}} \in EWMON^d-TXT$.*

The next result sheds some light on the power of class comprising dual strong-monotonic inference. By definition, a dual strong-monotonic IIM M might initially output a most general hypothesis. Afterwards, M might specialize its actual hypothesis until it eventually reaches a correct guess. As we have seen in the demonstration of Theorem 6, M, when outputting once an overgeneralized language, definitely fails to dual strong-monotonically learn this language itself. Thus, the potential strength of this approach may only come into the game, if class comprising hypothesis spaces are admissible (cf. Lange and Zeugmann [35]).

Theorem 29. $CSMON^d-TXT \;\#\; CSMON-TXT$

Proof. First, we show $CSMON-TXT \setminus CSMON^d-TXT \neq \emptyset$. As we have already seen, the family $\mathcal{L}_{finvar}$, that is, the family defined in Example 1, can be strong-monotonically inferred from text. On the other hand, by Lemma 1 below one immediately gets $\mathcal{L}_{finvar} \notin CSMON^d-TXT$.

***Lemma* 1.** *Let $\mathcal{L} = (L_j)_{j \in \mathbb{N}}$ be an indexed family. If $\mathcal{L} \in CSMON^d-TXT$, then either $L_k = L_j$ or $L_k \;\#\; L_j$ for all $j,\ k \in \mathbb{N}$.*

Proof. Suppose the converse, i.e., there is an indexed family $\mathcal{L} \in CSMON^d-TXT$ such that $L_j \subset L_k$ for some $j,\ k \in \mathbb{N}$. Let M denote any IIM inferring $\mathcal{L}$ dual strong-monotonically with respect to a hypothesis space $\mathcal{G}$ comprising $\mathcal{L}$. Moreover, let t be any text for L_j. Since M has to infer L_j from t, there has to be an $x \in \mathbb{N}$ such that $M(t_x) = j_x$ and $L(G_{j_x}) = L_j$. Since $L_j \subset L_k$, t_x can be extended to a text $\hat{t}$ for L_k. It is easy to see that M fails to infer L_k from $\hat{t}$ dual strong-monotonically. This proves Lemma 1.

We continue in showing $CSMON^d-TXT \setminus CSMON-TXT \neq \emptyset$. The desired indexed family is a slightly modified version of that one introduced in Example 3. For all $k \in \mathbb{N}$, we set $L_{\langle k,0\rangle} = \{a^k b^n \mid n \in \mathbb{N}^+\}$. For all $k \in \mathbb{N}$ and all $j > 1$, we distinguish the following cases:

Case 1. $\neg\ \Phi_k(k) \leq j$

We set $L_{\langle k,j\rangle} = L_{\langle k,0\rangle}$.

Case 2. $\Phi_k(k) \leq j$

Let $d = 2 \cdot \Phi_k(k) - j$. Then, we set:

$$L_{\langle k,j\rangle} = \begin{cases} \{a^k b^m \mid m \leq d\} \cup \{b^k a^j\}, \text{ if } d \geq 1, \\ \{a^k b\} \cup \{b^k a^j\}, \qquad\qquad \text{otherwise.} \end{cases}$$

Obviously, $\mathcal{L} = (L_{\langle k,j\rangle})_{j,k\in\mathbb{N}}$ is an indexed family of recursive languages.

***Lemma* 2.** $\mathcal{L} \notin CSMON{-}TXT$

This lemma can be proved by reducing the halting problem to $\mathcal{L} \in CSMON{-}TXT$. Thereby, the same construction outlined in the proof of Theorem 4 applies *mutatis mutandis*.

***Lemma* 3.** $\mathcal{L} \in CSMON^d{-}TXT$

We have to show that there are an appropriate hypothesis space $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ comprising $\mathcal{L}$ and an IIM M inferring $\mathcal{L}$ dual strong-monotonically with respect to $\mathcal{G}$.

We define the wanted hypothesis space $\mathcal{G}$ as follows. For all $k, j \in \mathbb{N}$, we set $L(G_{\langle k,0\rangle}) = \bigcup_{j\in\mathbb{N}} L_{\langle k,j\rangle}$ and $L(G_{\langle k,j+1\rangle}) = L_{\langle k,j\rangle}$.

Since $\mathcal{L}$ is an indexed family, it is easy to verify that membership is uniformly decidable for $\mathcal{G}$. By construction, $\mathcal{G}$ comprises $\mathcal{L}$. It remains to show that $\mathcal{L} \in CSMON^d{-}TXT$.

The IIM M defined below dual strong-monotonically infers $\mathcal{L}$ with respect to $\mathcal{G}$. Let $L \in \mathcal{L}$, let $t \in text(L)$, and let $x \in \mathbb{N}$. By construction, there is a unique k such that $t_x^+ \subseteq L_{\langle k,j\rangle}$ for some $j \in \mathbb{N}$. Clearly, the first string in t_x can be used to determine k.

IIM *M*: "On input t_x do the following: Test whether or not $\Phi_k(k) \leq x$. In case it is, goto (A). Otherwise, output $\langle k, 0\rangle$ and request the next input.

(A) Test whether or not $a^k b^{\Phi_k(k)+m} \in t_x^+$ for some $m \in \mathbb{N}^+$. In case it is, output $\langle k, 1\rangle$ and request the next input. Otherwise, goto (B).

(B) Test whether or not $b^k a^j \in t_x^+$ for some $j \in \mathbb{N}$. In case it is, output $\langle k, j+1\rangle$ and request the next input.
Otherwise, output nothing and request the next input."

Due to the definition above, M performs at most one mind change when inferring L from text t. Due to the definition of $L(G_{\langle k,0\rangle})$, we have $L_{\langle k,j\rangle} \subseteq L(G_{\langle k,0\rangle})$ for all $j \in \mathbb{N}$. This guarantees that this mind change fulfills the dual strong-monotonicity requirement. Finally, it is easy to verify that M converges to a correct hypothesis for L. □

Analogously to the class preserving case, $CFIN{-}TXT \subset CSMON{-}TXT$ can be easily shown. Hence, we obtain the following corollary which points to another difference between class preserving and class comprising learning (cf. Lange and Zeugmann [34, 35]).

Corollary 30. $CFIN{-}TXT \subset CSMON^d{-}TXT$

The additional power of a dual strong-monotonic IIM results from the fact that it can successfully exploit its ability to output overgeneralized hypotheses. On the other hand, it can be straightforwardly verified that even in the class comprising case weak-monotonic learners are exactly as powerful as conservatively working IIMs. This may lead to the impression that the relation between weak-monotonic and dual strong-monotonic inference will also be influenced. Lange and Zeugmann [34, 35] have shown that this is not the case.

Theorem 31. $CSMON^d\!-\!TXT \subset EWMON\!-\!TXT$

Proof. Let $\mathcal{L} \in CSMON^d\!-\!TXT$. The wanted IIM M is defined as follows. Let $L \in \mathcal{L}$, let $t \in text(L)$, and let $x \in \mathbb{N}$. We set:

IIM *M*: "On input t_x do the following: Search the least index $j \le x$ with $t_x^+ \subseteq L_j$. Output it. In case $t_x^+ \not\subseteq L_j$ for all $j \le x$ output nothing and request the next input."

By construction, M performs exclusively justified mind changes. Let $z = min\{j \mid L = L_j\}$. By Lemma 1 in Theorem 29 we may conclude that $L_z \setminus L_k \neq \emptyset$ for all $k < z$. Consequently, for all $k < z$ there has to be an x_k such that $t_{x_k}^+ \not\subseteq L_k$. Therefore, M rejects any hypothesis k with $k < z$ and converges to z. □

Next, we discuss the relation between monotonic and weak–monotonic inference. As a matter of fact notice that the family $\mathcal{L}_{wmon}$ introduced in the proof of Theorem 16 witnesses $CWMON\!-\!TXT \setminus CMON\!-\!TXT \neq \emptyset$, too. However, the proof technique developed in proving Theorem 1 cannot be applied to establish the inclusion $CMON\!-\!TXT \subseteq CWMON\!-\!TXT$. Moreover, looking at the following theorem, this inclusion though imaginable might even seem rather unlikely, since class comprising monotonic learning achieves its full power by possible overgeneralization. Hence, $CMON\!-\!TXT$ achieves its full learning capabilities by realizing a suitable interplay between generalization and specialization.

Theorem 32. *There is an indexed family $\mathcal{L} \in CMON\!-\!TXT$ such that there does not exist any conservative IIM M witnessing $\mathcal{L} \in CMON\!-\!TXT$.*

Proof. The desired indexed family is defined as follows. Initially, we set $L_0 = \{a\}^+$. For all $k \in \mathbb{N}$, we set $L_{2k+1} = \{a^k b\} \cup \{a^m \mid 1 \le m \le \Phi_k(k)\} \cup \{a^{\Phi_k(k)+10}\}$ and $L_{2k+2} = \{a^k b\} \cup \{a^m \mid 1 \le m \le \Phi_k(k)\} \cup \{a^{\Phi_k(k)+20}\}$. If $\varphi_k(k)$ is undefined, than $\{a^{\Phi_k(k)+10}\}$ and $\{a^{\Phi_k(k)+20}\}$, respectively, are used as a shorthand for the empty set. Consequently, $\varphi_k(k)\uparrow$ implies $L_{2k+1} = L_{2k+2} = \{a^k b\} \cup \{a\}^+$. By definition of a complexity measure we can easily conclude that $\mathcal{L} = (L_j)_{j\in\mathbb{N}}$ constitutes an indexed family.

First, we show $\mathcal{L} \in CMON\!-\!TXT$. The wanted space of hypotheses $\mathcal{G} = (G_j)_{j\in\mathbb{N}}$ is defined as follows. Let $L(G_0) = \{a\}^+$. For all $k \in \mathbb{N}$, set $L(G_{3k+1}) = L_{2k+1}$, $L(G_{3k+2}) = L_{2k+2}$, and $L(G_{3k+3}) = L_{2k+1} \cup L_{2k+2}$.

Assume any $L \in \mathcal{L}$, any text t for L, and any $x \in \mathbb{N}$. The desired monotonic IIM M works as follows:

IIM *M*: "On input t_x do the following: If $t_x^+ \subseteq \{a\}^+$, output 0 and request the next input. Otherwise, goto (A).

(A) Determine the unique k such that $a^k b \in t_x^+$. Test whether or not $\Phi_k(k) \le x$. In case it is not, output $3k+3$ and request the next input. Otherwise, goto (B).

(B) Let $y = \Phi_k(k)$. If the string a^{y+10} appears in t_x, output $3k+1$ and request the next input.
If the string a^{y+20} has been presented, output $3k+2$ and request the next input.
Otherwise, output $3k+3$ and request the next input."

Now, it is not hard to verify that M learns L. It remains to show that M works monotonically. Obviously, if $L = \{a\}^+$, then M never changes its mind. Thus, M monotonically infers L. Finally, let $k \in \mathbb{N}$ and $L \subseteq \{a^k b\} \cup \{a\}^+$. We distinguish the following cases:

Case 1. $\varphi_k(k)$ is undefined.

By definition of $\mathcal{L}$ we obtain $L = \{a^k b\} \cup \{a\}^+ = L_{2k+1} = L_{2k+2}$. Since $L(G_{3k+3}) = L_{2k+1} \cup L_{2k+2} = L$, M changes its mind at most one time and, therefore, it behaves monotonically.

Case 2. $\varphi_k(k)$ is defined.

Suppose that $L = L_{2k+1}$. By construction, M converges to the correct guess $3k+1$. In worst case, it may happen that M outputs subsequently the guesses 0, $3k+3$, and $3k+1$. By definition $L(G_{3k+3}) \cap L_{2k+1} = L_{2k+1}$. Therefore, $L(G_{3k+3}) \cap L_{2k+1} = L(G_{3k+1}) \cap L_{2k+1}$ which guarantees that M works monotonically. Note that the subcase $L = L_{2k+2}$ can be handled in a similar manner.

Consequently, M monotonically infers $\mathcal{L}$ with respect to $\mathcal{G}$.

As we have seen, M has the freedom to produce overgeneralized hypotheses. Next, we show that in order to learn $\mathcal{L}$ monotonically it is inevitable to allow overgeneralization.

Suppose the converse, that is, there are an IIM M and a class comprising space of hypotheses $\mathcal{G}$ for $\mathcal{L}$ such that M monotonically as well as conservatively infers $\mathcal{L}$ with respect to $\mathcal{G}$. We shall show that M may be used to decide the halting problem.

Claim. Given any program for M and $\mathcal{G}$, one may *non-effectively* obtain an algorithm deciding "$\varphi_k(k) \downarrow$."

By assumption M, in particular, infers the language $\{a\}^+$ from text $t = a, a^2, a^3, \dots$ Hence, there exists a $z \in \mathbb{N}$ such that $M(t_z) = j$ and $L(G_j) = \{a\}^+$. Given this index z we can define the following algorithm $\mathcal{A}$.

Algorithm $\mathcal{A}$: "On input k execute (A) and (B).

(A) Test whether or not $\Phi_k(k) \le z+1$. In case it is, output "$\varphi_k(k) \downarrow$." Otherwise, goto (B).

(B) We set $\hat{t} = \underbrace{a,\ \dots,\ a^{z+1}}_{=t_z},\ a^k b,\ a^{z+2},\ a^{z+3},\ \dots$ For $m = 2,\ 3,\ \dots$ execute in parallel (β1) and (β2) until (β3) or (β4) happens.

(β1) Test whether $\Phi_k(k) = z+m$.

(β2) Compute $j_{z+m} = M(\hat{t}_{z+m})$.

(β3) $\Phi_k(k) = z+m$ is verified. Then output "$\varphi_k(k) \downarrow$."

($\beta 4$) In ($\beta 2$) a hypothesis $j_{z+m} = M(\hat{t}_{z+m})$ is computed that satisfies $a^k b \in L(G_{j_{z+m}})$. Then output "$\varphi_k(k)\uparrow$" and stop."

Applying the same argument as in the proof of Theorem 9, Lemma 3, one can show that $\mathcal{A}$ terminates on every input k. To see this one has to take into consideration that "$\varphi_k(k)\uparrow$" implies that $L = \{a^k b\} \cup \{a\}^+$. Since, M has to infer L on its text $\hat{t}$, ($\beta 4$) happens.

It remains to show that $\mathcal{A}$ works correctly. Obviously, this reduces to the question whether $\varphi_k(k)$ is indeed undefined, if ($\beta 4$) happens. Suppose to the contrary that ($\beta 4$) happens, but $\varphi_k(k)$ is defined. Let $\Phi_k(k) = y$. By construction, we get $y > z + m$. Taking $\mathcal{L}$'s definition into consideration we obtain $L_{2k+1} \neq L_{2k+2}$. Furthermore, one can easily verify that $\hat{t}_{z+m}$ is an initial segment of a text for L_{2k+1} as well as of a text for L_{2k+2}. Recall that M has already generated two different guesses j and j_{z+m} when fed $\hat{t}_{z+m}$ where $L(G_j) = \{a\}^+$. Because M monotonically infers both L_{2k+1} and L_{2k+2} from texts having the prefix $\hat{t}_{z+m}$, we may conclude that $L(G_j) \cap L_{2k+1} \subseteq L(G_{j_{z+m}}) \cap L_{2k+1}$ as well as $L(G_j) \cap L_{2k+2} \subseteq L(G_{j_{z+m}}) \cap L_{2k+2}$. By definition of $\mathcal{L}$ we obtain $L(G_j) \cap L_{2k+1} = L_{2k+1} \setminus \{a^k b\}$ and $L(G_j) \cap L_{2k+2} = L_{2k+2} \setminus \{a^k b\}$. Since $a^k b \in L(G_{j_{z+m}})$, we conclude $L(G_{j_{z+m}}) \supseteq L_{2k+1} \cup L_{2k+2}$. Finally, since $L_{2k+1} \neq L_{2k+2}$, it follows $L(G_{j_{z+m}}) \supset L_{2k+1}$, for instance. Hence, M has generated an overgeneralized hypothesis when fed any text $\tilde{t}$ for L_{2k+1} having the prefix $\hat{t}_{z+m}$. This contradicts our assumption that M works in a conservative fashion. This proves the claim.

Finally, taking into account that the halting problem is undecidable we may conclude that overgeneralization is inevitable in order to learn $\mathcal{L}$ in a monotonic fashion. $\square$

Since any class preservingly working monotonic IIM has, in general, to avoid overgeneralization (cf. Theorem 1), we immediately get the following corollary.

Corollary 33. $MON{-}TXT \subset CMON{-}TXT$

On the other hand, one can easily observe that the indexed family $\mathcal{L}$ used in the above proof belongs to $EWMON{-}TXT$. Clearly, this cannot be true in general as Theorem 3 impressively witnesses. On the other hand, non–monotonic cumulative reasoning can be successfully exploited in the learning process provided that overgeneralization can be algorithmically avoided. But the capability of any IIM to algorithmically avoid overgeneralization mainly depends on the degree of recursiveness the relevant hypothesis space possesses. Consequently, the problem whether or not $CMON{-}TXT \subseteq CWMON{-}TXT$ mainly reduces itself to the following question: Does there exist a universal method to design the relevant hypothesis space in a way such that overgeneralization can be effectively prevented. This is indeed the case as our next theorem shows.

Theorem 34. $CMON{-}TXT \subset CWMON{-}TXT$.

Proof. As already mentioned $CWMON{-}TXT \setminus CMON{-}TXT \neq \emptyset$ is witnessed by the family $\mathcal{L}_{wmon}$ introduced in the proof of Theorem 16. It remains to show that $CMON{-}TXT \subseteq CWMON{-}TXT$.

Let $\mathcal{L} \in CMON{-}TXT$. Moreover, let $\hat{M}$ be a monotonic IIM which infers $\mathcal{L}$ with respect to a class comprising hypothesis space $\hat{\mathcal{G}} = (\hat{G}_j)_{j\in\mathbb{N}}$. Without loss of generality we may assume that $\hat{M}$ outputs in every step a hypothesis.

Next, we define a hypothesis space $\mathcal{G} = (G_{\langle j,k\rangle})_{j,k\in\mathbb{N}}$ which allows a weak–monotonic learner to avoid overgeneralization when inferring $\mathcal{L}$ with respect to $\mathcal{G}$. For all $j,\ k \in \mathbb{N}$, we set $L(G_{\langle j,k\rangle}) = L(\hat{G}_j) \cap L_k$. Obviously, $\mathcal{G}$ defines a class comprising hypothesis space for $\mathcal{L}$.

Claim. Let $L \in \mathcal{L}$, $t \in text(L)$, and $j \in \mathbb{N}$ any hypothesis $\hat{M}$ has generated when fed t. Then, for all $k \in \mathbb{N}$, we have $L \not\subset L(G_{\langle j,k\rangle})$.

Suppose to the contrary that there is an initial segment of t, say t_x, such that $\hat{M}(t_x) = j$ and $L \subset L(G_{\langle j,k\rangle})$ for some $k \in \mathbb{N}$. By definition of $\mathcal{G}$ we obtain $L \subset L(\hat{G}_j)$ as well as $L \subset L_k$. Having this in mind we show that there is a text $\hat{t}$ for L_k such that $\hat{M}$ violates the monotonicity constraint when inferring L_k from this particular text. Since $\hat{M}$ infers L, there has to be a $y > x$ such that $\hat{M}(t_y) = z$ with $L = L(\hat{G}_z)$. Obviously, t_y can be extend to become a text, say $\hat{t}$, for L_k. By construction $\hat{M}$ has subsequently output the guesses j and z when fed $\hat{t}$. Moreover, $L \subset L(\hat{G}_j) \cap L_k$, but $L = L(\hat{G}_z) \cap L_k$, since $L = L(\hat{G}_z)$. Therefore, $L(\hat{G}_j) \cap L_k \not\subset L(\hat{G}_z) \cap L_k$. This contradicts our assumption that $\hat{M}$ monotonically infers $\mathcal{L}$ with respect to the hypothesis space $\hat{\mathcal{G}}$. Hence, the claim is proved.

Finally, we define an IIM M which weak–monotonically learns $\mathcal{L}$ with respect to the new hypothesis space $\mathcal{G}$. Let $L \in \mathcal{L}$, $t \in text(L)$, and $x \in \mathbb{N}$.

IIM *M*: "On input t_x do the following: If $x = 0$, then compute $\hat{M}(t_0) = j$ and output the guess $\langle j, 0\rangle$. Otherwise, execute Instruction (A).

(A) Let $z = M(t_{x-1})$. If $t_x^+ \subseteq L(G_z)$, then repeat the guess z, and request the next input.
Otherwise, compute $n = \hat{M}(t_x)$ and determine the least index k such that $t_x^+ \subseteq L_k$. Output the hypothesis $\langle n, k\rangle$ and request the next input."

By definition M performs exclusively justified mind changes. Hence, M behaves weak-monotonically. Furthermore, since $L \in \mathcal{L}$ and t is a text for L, it is easy to verify that the *unbounded* search within Instruction (A) terminates for every initial segment t_x. Thus, M is indeed a weak–monotonic IIM.

Now, we have to argue that M converges to a correct hypothesis for L. This can be easily verified, if one additionally takes into consideration that M never outputs an overgeneralized hypothesis (cf. the above claim) and that the IIM $\hat{M}$ converges to a correct hypothesis for L. We omit further details. $\square$

Finally, let us mention that all other relations between the different notions of monotonicity and dual monotonicity remain unchanged, if class comprising learning from text is investigated. Furthermore, a comparison between class comprising monotonic and dual-monotonic learning from informant results in the same overall picture as in the class preserving case.

7. Conclusions and Open Problems

We have studied the power of strong-monotonic, monotonic, and weak-monotonic as well as dual strong-monotonic, dual monotonic, and dual weak-monotonic IIMs in the setting of algorithmic language learning from both only positive as well as positive and negative examples. These constraints had been proposed to help to provide understanding of the utility and necessity of non-monotonic reasoning in the process of learning in general. Monotonicity constraints in the most general sense enforce a discipline on the learning process, requiring that the hypotheses that the learner produces 'improve' in this or that sense monotonically. We have related the various possible notions to each other and, as a result, obtained several new hierarchies. In order to broaden our understanding even further, we decided to consider the entire range of possible relations between the hypothesis space and the language family to be learned. In the order of decreasing severity of restriction, we considered exact learning, where the two have to be the exactly the same, class-preserving learning, where they have to be semantically the same but can be syntactically different, and finally, class-comprising learning, where the minimal constraint, that the hypothesis space contains all the languages to be learned, is enforced. A number of our results show that with any type of monotonic or dual monotonic inference, this choice of the hypothesis space is of great influence. In general, when dealing with any type of monotonic or dual monotonic learning, *exactness* and *class preserveness* considerably restricts the inference power. Furthermore, one of our separation results is heavily dependent on *class-preserving* hypothesis spaces, i.e., $SMON^d{-}TXT \subseteq FIN{-}TXT$. Moreover, we have shown that indeed $CSMON^d{-}TXT$ becomes more powerful than $CFIN{-}TXT$. In fact, it turns out that $CSMON^d{-}TXT \# CSMON{-}TXT$. Some other recent work has also systematically studied the impact of different requirements on the admissible spaces of hypotheses (Lange and Zeugmann [34, 35]).

Even in the case of class comprising learning, our results show that (dual) strong-monotonicity and (dual) monotonicity lead to severe restrictions on the inference power. This means that situations in which learning processes have to perform non-monotonically or at best weak-monotonically or dual weak-monotonically are likely to arise.

It remains a challenge to combine monotonic language learning with other postulates of naturalness such as iterative inference introduced by Wiehagen [48] or order independence (cf. Fulk [14], and the references therein). Various results along this line may be found in Lange and Zeugmann [32, 36] and Zeugmann and Lange [52]. Moreover, it seems very interesting to study the learnability of indexed families by *probabilistic* IIMs.

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. We heartily thank the anonymous referees whose comments improved the paper immensely.

8. References

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

[2] D. Angluin, Finding patterns common to a set of strings, *Journal of Computer and System Sciences* **21** (1980) 46–62.

[3] D. Angluin and C. H. Smith, Inductive inference: theory and methods, *Computing Surveys* **15** (1983) 237–269.

[4] D. Angluin and C. H. Smith, Formal inductive inference, in: *Encyclopedia of Artificial Intelligence, Vol. 1* (Wiley-Interscience Publication, New York, 1987) 409–418.

[5] S. Arikawa, S. Goto, S. Ohsuga and T. Yokomori (eds.), *Proc. of the 1st International Workshop on Algorithmic Learning Theory*, (Japanese Society for Artificial Intelligence, Tokyo, 1990).

[6] S. Arikawa, A. Maruoka and T. Sato (eds.), *Proc. of the 2nd International Workshop on Algorithmic Learning Theory*, (Japanese Society for Artificial Intelligence, Tokyo, 1991).

[7] Ya. M. Barzdin, Inductive inference of automata, functions and programs, in: *Proc. of the International Congress of Mathematics* (Vancouver, 1974) 455–460.

[8] H. R. Beick, Personal Communication, 1991.

[9] L. Blum and M. Blum, Toward a mathematical theory of inductive inference, *Information and Control* **28** (1975), 122–155.

[10] A. Blum and M. Singh, Learning functions of k terms, in: *Proc. of the 3rd Workshop on Computational Learning Theory* (Morgan Kaufmann Publishers Inc., San Mateo, 1990) 144–153.

[11] G. Brewka, *Nonmonotonic Reasoning: Logical Foundations of Commonsense*, (Cambridge University Press, Cambridge, 1991).

[12] J. Case, The power of vacillation, in: *Proc. of the 1st Workshop on Computational Learning Theory* (Morgan Kaufmann Publishers Inc., San Mateo, 1988) 196–205.

[13] J. Case and C. Lynes, Machine inductive inference and language identification, in: *Proc. of the Ninth International Conference on Automata, Languages and Programming* Lecture Notes in Computer Science **140** (Springer, Berlin, 1982) 107–115.

[14] M. Fulk, Prudence and other restrictions in formal language learning, *Information and Computation* **85** (1990) 1–11.

[15] M. Fulk and J. Case (eds.), *Proc. of the 3rd Annual Workshop on Computational Learning Theory*, (Morgan Kaufmann Publishers Inc., San Mateo, 1990).

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

[17] D. Haussler (ed.), *Proc. of the 5th Annual Workshop on Computational Learning Theory* (ACM Press, New York, 1992).

[18] J. E. Hopcroft and J. D. Ullman, *Formal Languages and their Relation to Automata* (Addison-Wesley, Reading, Massachusetts, 1979).

[19] S. Jain and A. Sharma, Recursion theoretic characterizations of language learning, The University of Rochester, Dept. of Computer Science, TR 281.

[20] K. P. Jantke (ed.), *Proc. of the 2nd International Workshop on Analogical and Inductive Inference*, Lecture Notes in Artificial Intelligence Vol. 397, (Springer-Verlag, Berlin, 1989).

[21] K. P. Jantke, Monotonic and non–monotonic inductive inference, *New Generation Computing* **8** (1991) 349–360.

[22] K. P. Jantke, Monotonic and non-monotonic inductive inference of functions and patterns, in: *Proc. of the 1st International Workshop on Nonmonotonic and Inductive Logic* Lecture Notes in Artificial Intelligence **543** (Springer-Verlag, Berlin, 1991) 161–177.

[23] K. P. Jantke (ed.), *Proc. of the 3rd International Workshop on Analogical and Inductive Inference*, Lecture Notes in Artificial Intelligence Vol. 642, (Springer-Verlag, Berlin, 1992).

[24] T. Jiang, A. Salomaa, K. Salomaa and S. Yu, Inclusion is undecidable for pattern languages, in: *Proc. 20th International Colloquium on Automata, Languages and Programming* Lecture Notes in Computer Science Vol. 700, (Springer, Berlin, 1993) 301–312.

[25] S. Kapur, Computational learning of languages, Ph.D. thesis, Cornell University, Computer Science Department Technical Report 91-1234, 1991.

[26] M. Kearns and L. Pitt, A polynomial-time algorithm for learning k–variable pattern languages from examples, in: *Proc. of the 2nd Annual Workshop on Computational Learning Theory* (Morgan Kaufmann Publishers Inc., San Mateo, 1989) 57 –71.

[27] K. Ko, A. Marron and W. G. Tzeng, Learning string patterns and tree patterns from examples, in: *Proc. of the 7th Conference on Machine Learning* (Morgan Kaufmann Publishers Inc., San Mateo, 1990) 384–391.

[28] Y. Kodratoff and R. S. Michalski, *Machine Learning, An Artificial Intelligence Approach, Vol. 3* (Morgan Kaufmann Publishers Inc., San Mateo, 1990).

[29] S. Lange, The representation of recursive languages and its impact on the efficiency of learning, in: *Proc. of the 7th Annual ACM Conference on Computational Learning Theory*, (ACM Press, New York, 1994) 256–267.

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

[31] S. Lange and T. Zeugmann, On the power of monotonic language learning, TH Leipzig, Fachbereich Mathematik und Informatik, GOSLER–Report 05/92, 1992.

[32] S. Lange and T. Zeugmann, Monotonic versus non-monotonic language learning, in: *Proc. of the 2nd International Workshop on Nonmonotonic and Inductive Logic* Lecture Notes in Artificial Intelligence **659** (Springer-Verlag, Berlin, 1993) 254–269.

[33] S. Lange and T. Zeugmann, Learning recursive languages with bounded mind changes, *International Journal of Foundations of Computer Science* **4** (1993), 157–178.

[34] S. Lange and T. Zeugmann, Language learning in dependence on the space of hypotheses, in: *Proc. of the 6th Annual ACM Conference on Computational Learning Theory* (ACM Press, New York, 1993) 127–136.

[35] S. Lange and T. Zeugmann, The learnability of recursive languages in dependence on the space of hypotheses, TH Leipzig, Fachbereich Mathematik und Informatik, GOSLER–Report 20/93, 1993.

[36] S. Lange and T. Zeugmann, On the impact of order independence to the learnability of recursive languages, FUJITSU Laboratories Ltd., Institute for Social Information Science, Numazu, Research Report ISIS-RR-93-17E, 1993.

[37] S. Lange and T. Zeugmann, Characterization of language learning on informant under various monotonicity constraints, *Journal of Experimental and Theoretical Artificial Intelligence* **6** (1994) 73–94.

[38] M. Machtey and P. Young, *An Introduction to the General Theory of Algorithms* (North-Holland, New York, 1978).

[39] R. S. Michalski, J. G. Carbonell and T. M. Mitchell, *Machine Learning, An Artificial Intelligence Approach, Vol. 1* (Springer-Verlag, Berlin, 1984).

[40] R. S. Michalski, J. G. Carbonell and T. M. Mitchell, *Machine Learning, An Artificial Intelligence Approach, Vol. 2* (Morgan Kaufmann Publishers Inc., San Mateo, 1986).

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

[42] 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).

[43] L. Pitt and L.G. Valiant, Computational limitations on learning from examples, *Journal of the ACM* **35**, 965–984.

[44] R. Rivest, D. Haussler and M.K. Warmuth (eds.), *Proc. of the 2nd Annual Workshop on Computational Learning Theory*, (Morgan Kaufmann Publishers Inc., San Mateo, 1989).

[45] T. Shinohara, Polynomial time inference of extended regular pattern languages, in: *Proceedings of the RIMS Symposia on Software Science and Engineering* Lecture Notes in Computer Science **147** (Springer-Verlag, Berlin, 1982) 115–127.

[46] R. Solomonoff, A formal theory of inductive inference, *Information and Control* **7** (1964) 1–22, 234–254.

[47] B. A. Trakhtenbrot and Ya. M. Barzdin, *Finite Automata–Behavior and Synthesis, Fundamental Studies in Computer Science 1* (North–Holland, Amsterdam, 1973).

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

[49] R. Wiehagen, Identification of formal languages, in: *Proc. Mathematical Foundations of Computer Science* Lecture Notes in Computer Science **53** (Springer-Verlag, Berlin, 1977) 571–579.

[50] R. Wiehagen, A thesis in inductive inference, in: *Proc. of the 1st International Workshop on Nonmonotonic and Inductive Logic* Lecture Notes in Artificial Intelligence **543** (Springer-Verlag, Berlin, 1991) 184–207.

[51] R. Wiehagen and T. Zeugmann, Ignoring data may be the only way to learn efficiently, *Journal of Experimental and Theoretical Artificial Intelligence* **6** (1994) 131–144.

[52] T. Zeugmann and S. Lange, A guided tour across the boundaries of learning recursive languages, in: *Algorithmic Learning for Knowledge-Based Systems* (K.P. Jantke and S. Lange, Eds.), Lecture Notes in Artificial Intelligence **961** (Springer-Verlag, Berlin, 1995) 190 – 258.

[53] T. Zeugmann, S. Lange and S. Kapur, Characterizations of monotonic and dual monotonic language learning, *Information and Computation***120** (1995) 155 – 173.