

U-shaped, iterative, and iterative-with-counter learning

John Case · Samuel E. Moelius III

Received: 31 August 2007 / Revised: 2 February 2008 / Accepted: 20 February 2008 /
Published online: 19 March 2008
Springer Science+Business Media, LLC 2008

Abstract This paper solves an important problem left open in the literature by showing that *U-shapes* are unnecessary in *iterative learning* from positive data. A *U-shape* occurs when a learner first *learns*, then *unlearns*, and, finally, *relearns*, some target concept. *Iterative learning* is a Gold-style learning model in which each of a learner's output conjectures depends *only* upon the learner's most recent conjecture and input element. Previous results had shown, for example, that U-shapes are unnecessary for explanatory learning, but *are* necessary for behaviorally correct learning.

Work on the aforementioned problem led to the consideration of an iterative-like learning model, in which each of a learner's conjectures may, *in addition*, depend upon the number of elements so far presented to the learner. Learners in this new model are strictly more powerful than traditional iterative learners, yet not as powerful as full explanatory learners. Can any class of languages learnable in this new model be learned without U-shapes? For now, *this* problem is left open.

Keywords Computational learning theory · Counter · Gold-style learning · Inductive inference · Iterative learning · Iterative-with-counter learning · Language learning · Memory limited learning · Non-U-shaped learning · Partly set-driven learning · Rearrangement independent learning · Set-driven learning · U-shape · U-shaped learning

1 Introduction

1.1 U-shapes

A *U-shape* occurs when a learner first *learns*, then *unlearns*, and, finally, *relearns*, some target concept. This phenomenon has been observed, for example, in children learning the

Editors: Claudio Gentile, Nader H. Bshouty.

This paper is an expanded version of (Case and Moelius 2007).

J. Case · S.E. Moelius III (✉)

Department of Computer and Information Science, University of Delaware, Newark, PA 19716, USA
e-mail: moelius@cis.udel.edu

use of regular and irregular verbs, e.g., a child first correctly learns that the past tense of “speak” is “spoke”; then, the child overregularizes and *incorrectly* uses “speaked”; finally, the child returns to correctly using “spoke” (Marcus et al. 1992; Plunkett and Marchman 1991; Taatgen and Anderson 2002).

Important questions regarding U-shapes are the following. Are U-shapes an *unnecessary* accident of human evolution, *or*, are there classes of tasks that can be learned *with* U-shapes, but *not* otherwise? That is, are there classes of tasks that are learnable *only* by returning to some *abandoned* correct behavior?

There have been mathematical attempts to answer these questions (Carlucci et al. 2006, 2007a, 2007b; Baliga et al. 2007) in the context of Gold-style language learning from positive data (Gold 1967; Jain et al. 1999).¹ Such models differ from one another in ways described hereafter, however, the following is common to all. Infinitely often, a *learner* is fed successively longer finite, initial sequences of an *infinite* sequence of numbers and, possibly, pauses (#). The set of all such numbers represents a *language*, and, the infinite sequence, itself, is called a *text* for the language. For each finite, initial sequence of a text, the learner either: outputs a *conjecture*, or diverges (e.g., goes into an infinite loop). A conjecture may be either: a grammar (possibly for the language represented by the text), or a ‘?’.

One way in which Gold models differ from one another is in the criteria used to judge the *success* of a learner. Examples of models with differing criteria are *explanatory learning* (**Ex**-learning) (Gold 1967; Jain et al. 1999)² and *behaviorally correct learning* (**Bc**-learning) (Case and Lynes 1982; Jain et al. 1999). In both models, for a learner to be successful, all but finitely many of the learner’s conjectures must correctly (semantically) identify the input language. However, **Ex**-learning has the additional requirement that a learner converge *syntactically* to a single conjecture.

In Gold-style learning, a U-shape is formalized as: outputting a semantically correct conjecture, then outputting a semantically incorrect conjecture, and, finally, returning to a semantically correct conjecture (Carlucci et al. 2006, 2007a, 2007b; Baliga et al. 2007). As it turns out, U-shapes are *unnecessary* for **Ex**-learning, i.e., every class of languages that can be **Ex**-learned can be **Ex**-learned *without* U-shapes (Baliga et al. 2007, Theorem 20). On the other hand, U-shapes *are* necessary for **Bc**-learning, i.e., there are classes of languages that *can* be **Bc**-learned *with* U-shapes, but *not* *without* (Fulk et al. 1994, proof of Theorem 4). Thus, in at least some contexts, this *seemingly* inefficient behavior can actually increase one’s learning power.³

1.2 Iterative learning

For both **Ex**-learning and **Bc**-learning, a learner is free to base a conjecture upon *every* element presented to the learner up to that point. Thus, in a sense, an **Ex**-learner or **Bc**-learner can *remember* every element presented to it. One could argue that such an ability is beyond that possessed by (most) humans. This calls into question the applicability of **Ex**-learning and **Bc**-learning to modeling human learning. That is, it would seem that any model of human learning should be *memory limited* in some respect.

¹In this paper, we focus exclusively on language learning from positive data. So, we do not consider, for example, function learning, nor language learning from informant (Jain et al. 1999).

²**Ex**-learning is the model that was actually studied by Gold (1967).

³There exist Gold models that lie strictly between **Ex** and **Bc** (Case 1999). For nearly every such model considered, U-shapes are necessary (Carlucci et al. 2007a).

Iterative learning (**It**-learning) (Wiehagen 1976; Lange and Zeugmann 1996a; Case et al. 1999) is a straightforward variation of the **Ex**-learning model that *is* memory limited.⁴ In this model, each of a learner's conjectures can depend *only* upon the learner's most recent conjecture and input element. An **It**-learner can remember elements fed to it by *coding* them into its conjectures. However, like an **Ex**-learner, an **It**-learner is required to converge syntactically to a single conjecture. Thus, on any given text, an **It**-learner can perform such a coding-trick for only finitely many elements.

Despite these apparent limitations, many useful classes of languages are **It**-learnable. For example, the *pattern languages* (Angluin 1980) are an **It**-learnable class of languages with applications to molecular biology (Arikawa et al. 1993; Shimozono et al. 1994; Shinohara and Arikawa 1995). (A pattern language is, by definition, the language generated by all positive length substitution instances in a *pattern*, e.g., $abXYcbbZXa$, where the variables/nonterminals are depicted in uppercase, and the constants/terminals are depicted in lowercase.) The pattern languages were first shown to be **It**-learnable by Lange and Wiehagen (1991). Subsequently, this result was extended by Case et al. (1999) who showed that, for each k , the class formed by taking the union of all choices of k pattern languages is **It**-learnable.

There have been previous attempts to determine whether U-shapes are necessary in **It**-learning (Jain 2006; Carlucci et al. 2007b). The memory limited aspect of **It**-learning makes it more nearly applicable than **Ex**-learning or **Bc**-learning to modeling human learning.

Herein (Theorem 2 in Sect. 3), we solve this important open problem by showing that U-shapes are *unnecessary* in **It**-learning, i.e., any class of languages that can be **It**-learned can be **It**-learned *without* U-shapes.

1.3 Other restricted forms of learning

Two other restricted forms of learning that have been well studied are *set-driven learning* (**SDEx**-learning) and *partly set-driven learning* (**PSDEx**-learning) (Wexler and Culicover 1980; Schäfer-Richter 1984; Fulk 1990; Lange and Zeugmann 1996b).⁵ The **SDEx**-learning model requires that a learner output syntactically identical conjectures when fed two different initial sequences with the same content, i.e., listing the same set of numbers. So, for example, when forming a conjecture, an **SDEx**-learner *cannot* consider the number of elements so far presented to it, or the order in which those elements were presented. The **PSDEx**-learning model is similar, except that a learner is required to output identical conjectures when fed initial sequences with the same content *and length*. Thus, when forming a conjecture, a **PSDEx**-learner *cannot* consider the order in which elements were presented to it, but *can* consider the number of such elements.

SDEx-learners and **It**-learners are alike in that neither can consider the number of elements so far presented to it when forming a conjecture. Furthermore, **PSDEx**-learners are like **SDEx**-learners with *just this one* restriction lifted. Herein, we consider a similar counterpart to **It**-learners. That is, we consider a model in which each of a learner's output conjectures can depend *only* upon the learner's most recent conjecture, the most recent input element, *and a counter* indicating the number of elements so far presented to the learner. For example, if the learner were fed the sequence

$$0, 1, \#, \#, 0,$$

⁴Other memory limited models are considered in (Osherson et al. 1986; Fulk et al. 1994; Case et al. 1999; Carlucci et al. 2007b).

⁵**PSDEx**-learning is also called *rearrangement independent learning* in the literature (e.g., Lange and Zeugmann 1996b).

then the value of the counter would be 5. Thus, a repetition or pause is treated just like any other element in determining the value of the counter. We call this model *iterative-with-counter* learning (**ItCtr**-learning). In Sect. 4, we show that **ItCtr**-learning and **SDEx**-learning are incomparable (Theorems 3 and 4), i.e., for each, there is a class of languages learnable by that one, but *not* the other. It follows that **ItCtr**-learning is strictly more powerful than **It**-learning, yet not as powerful as full **Ex**-learning.

In an early attempt at showing that U-shapes are unnecessary in **It**-learning, we obtained the partial result that U-shapes are unnecessary in **It**-learning of classes of *infinite* languages. Independently, Sanjay Jain obtained the same (partial) result (Jain 2006). Thus, we hypothesize: learning without U-shapes is easier when the learner has access to some source of *infinite*, e.g., the cardinality of the input language. This belief is what led us to consider the **ItCtr**-learning model, as every learner in this model has access to a source of infinite, i.e., the counter, even when fed a text for a *finite* language.

Assuming our above hypothesis is correct, it should be easy to show that U-shapes are unnecessary in **ItCtr**-learning. Unfortunately, however, this problem has turned out to be more difficult than we had anticipated. So, for now, it is left open.

1.4 Organization

The remainder of this paper is organized as follows. Section 2, just below, gives notation and preliminaries. Section 3 proves our main result, namely, that U-shapes are *unnecessary* in **It**-learning. Section 4 explores **ItCtr**-learning, and, restates, formally, the problem that this paper leaves open. Section 5 concludes.

2 Notation and preliminaries

Computability-theoretic concepts not explained below are treated in (Rogers 1967). In matters of notation and terminology, we have, to a large extent, tried to be consistent with (Jain et al. 1999).

\mathbb{N} denotes the set of natural numbers, $\{0, 1, 2, \dots\}$. $\mathbb{N}_? \stackrel{\text{def}}{=} \mathbb{N} \cup \{?\}$. $\mathbb{N}_\# \stackrel{\text{def}}{=} \mathbb{N} \cup \{\#\}$. Lowercase Roman letters, with or without decorations, range over elements of \mathbb{N} , unless stated otherwise. A and L , with or without decorations, range over subsets of \mathbb{N} . \mathcal{L} ranges over collections of subsets of \mathbb{N} . For all A , $|A|$ denotes the cardinality of A . For all finite, nonempty A , $\max A$ denotes the maximum element of A . $\max \emptyset \stackrel{\text{def}}{=} -1$.

ψ ranges over one-argument partial functions from \mathbb{N} to \mathbb{N} . For all ψ and x , $\psi(x)\downarrow$ denotes that $\psi(x)$ converges; $\psi(x)\uparrow$ denotes that $\psi(x)$ diverges.⁶ For all ψ , $\text{dom}(\psi) \stackrel{\text{def}}{=} \{x : \psi(x)\downarrow\}$ and $\text{rng}(\psi) \stackrel{\text{def}}{=} \{y : (\exists x)[\psi(x) = y]\}$. We use \uparrow to denote the value of a divergent computation. λ denotes the empty function, i.e., the everywhere divergent function.

$\varphi_0, \varphi_1, \dots$ denotes any fixed, acceptable numbering of all one-argument partial computable functions from \mathbb{N} to \mathbb{N} (Rogers 1967). Φ denotes a fixed Blum complexity measure for φ (Blum 1967). For all p , $W_p \stackrel{\text{def}}{=} \text{dom}(\varphi_p)$. Thus, for all p , W_p is the p th recursively enumerable set (Rogers 1967). $W_\uparrow \stackrel{\text{def}}{=} \emptyset$.

⁶For all one-argument partial functions ψ and x , $\psi(x)$ converges iff there exists y such that $\psi(x) = y$; $\psi(x)$ diverges iff there is *no* y such that $\psi(x) = y$. If ψ is partial computable, and x is such that $\psi(x)$ diverges, then one can imagine that a program associated with ψ goes into an infinite loop on input x .

$\mathbb{N}_\#^*$ denotes the set of all finite initial segments of total functions of type $\mathbb{N} \rightarrow \mathbb{N}_\#$. $\mathbb{N}_\#^{\leq\omega}$ denotes the set of *all* (finite and infinite) initial segments of total functions of type $\mathbb{N} \rightarrow \mathbb{N}_\#$.⁷ $\alpha, \beta, \varrho, \sigma$, and τ , with or without decorations, range over elements of $\mathbb{N}_\#^*$. T , with or without decorations, ranges over total functions of type $\mathbb{N} \rightarrow \mathbb{N}_\#$.

For all $f \in \mathbb{N}_\#^{\leq\omega}$, $\text{content}(f) \stackrel{\text{def}}{=} \text{rng}(f) - \{\#\}$. For all T and L , T is a text for $L \stackrel{\text{def}}{\iff} \text{content}(T) = L$. For all σ , $|\sigma|$ (pronounced: the *length* of σ) $\stackrel{\text{def}}{=} |\text{dom}(\sigma)|$. For all $f \in \mathbb{N}_\#^{\leq\omega}$, and all σ, n , and i , $f[n](i)$ and $(\sigma \diamond f)(i)$ are defined as follows.

$$f[n](i) \stackrel{\text{def}}{=} \begin{cases} f(i), & \text{if } i < n; \\ \uparrow, & \text{otherwise.} \end{cases} \tag{1}$$

$$(\sigma \diamond f)(i) \stackrel{\text{def}}{=} \begin{cases} \sigma(i), & \text{if } i < |\sigma|; \\ f(i - |\sigma|), & \text{otherwise.} \end{cases} \tag{2}$$

\mathbf{M} , with or without decorations, ranges over partial computable functions of type $\mathbb{N}_\#^* \rightarrow \mathbb{N}_\#$.⁸

Definition 1, just below, introduces the Gold-style learning models considered in this paper. In this definition, **Ex**, **SD**, **PSD**, **It**, and **ItCtr** are mnemonic for *explanatory*, *set-driven*, *partly set-driven*, *iterative*, and *iterative-with-counter*, respectively.

Definition 1 For all \mathbf{M} and L , (a) through (e) below.

- (a) (Gold 1967) **M Ex-identifies** $L \iff$ for all texts T for L , there exist i and p such that $(\forall j \geq i)[\mathbf{M}(T[j]) = p]$ and $W_p = L$.
- (b) (Wexler and Culicover 1980) **M SDEx-identifies** $L \iff$ **M Ex-identifies** L , and, for all ϱ and σ , if $\text{content}(\varrho) = \text{content}(\sigma)$, then $\mathbf{M}(\varrho) = \mathbf{M}(\sigma)$.
- (c) (Schäfer-Richter 1984 and Fulk 1990) **M PSDEx-identifies** $L \iff$ **M Ex-identifies** L , and, for all ϱ and σ , if $|\varrho| = |\sigma|$ and $\text{content}(\varrho) = \text{content}(\sigma)$, then $\mathbf{M}(\varrho) = \mathbf{M}(\sigma)$.
- (d) (Wiehagen 1976) **M It-identifies** $L \iff$ **M Ex-identifies** L , and, for all ϱ, σ , and τ such that $\text{content}(\varrho) \cup \text{content}(\sigma) \cup \text{content}(\tau) \subseteq L$, (i) and (ii) below.
 - (i) $\mathbf{M}(\varrho) \downarrow$.
 - (ii) $\mathbf{M}(\varrho) = \mathbf{M}(\sigma) \implies \mathbf{M}(\varrho \diamond \tau) = \mathbf{M}(\sigma \diamond \tau)$.
- (e) **M ItCtr-identifies** $L \iff$ **M Ex-identifies** L , and, for all ϱ, σ , and τ such that $\text{content}(\varrho) \cup \text{content}(\sigma) \cup \text{content}(\tau) \subseteq L$, (i) and (ii) below.
 - (i) $\mathbf{M}(\varrho) \downarrow$.
 - (ii) $[|\varrho| = |\sigma| \wedge \mathbf{M}(\varrho) = \mathbf{M}(\sigma)] \implies \mathbf{M}(\varrho \diamond \tau) = \mathbf{M}(\sigma \diamond \tau)$.

Definition 2 For all $\mathcal{I} \in \{\mathbf{Ex}, \mathbf{SDEx}, \mathbf{PSDEx}, \mathbf{It}, \mathbf{ItCtr}\}$, (a) and (b) below.

- (a) For all \mathbf{M} , $\mathcal{I}(\mathbf{M}) = \{L : \mathbf{M} \mathcal{I}\text{-identifies } L\}$.
- (b) $\mathcal{I} = \{\mathcal{L} : (\exists \mathbf{M})[\mathcal{L} \subseteq \mathcal{I}(\mathbf{M})]\}$.

⁷A total function of type $\mathbb{N} \rightarrow \mathbb{N}_\#$ has exactly one infinite initial segment, namely, itself.

⁸Such an \mathbf{M} is often called an *inductive inference machine* (Jain et al. 1999). Note that since \mathbf{M} ranges over *partial* computable functions, it is permitted that \mathbf{M} diverge on elements of $\mathbb{N}_\#^*$. In particular, it may be the case that, for some σ and τ , $\mathbf{M}(\sigma) \uparrow$, yet $\mathbf{M}(\sigma \diamond \tau) \downarrow$. Also note that, for any **Ex**-identifiable class of languages \mathcal{L} (see Definition 1(a)), there exists a *total* learner \mathbf{M} such that \mathbf{M} **Ex**-identifies \mathcal{L} (Jain et al. 1999, Proposition 4.15).

Definition 3, just below, formalizes the notion of *U-shape*. In part (b) of this definition, **NU** is mnemonic for *non-U-shaped*.

Definition 3 For all $\mathcal{I} \in \{\mathbf{Ex}, \mathbf{SDEx}, \mathbf{PSDEx}, \mathbf{It}, \mathbf{ItCtr}\}$, (a) and (b) below.

- (a) For all \mathbf{M} , L , and texts T for L , \mathbf{M} exhibits a *U-shape* on $T \Leftrightarrow$ there exist i, j , and k such that $i < j < k$, $\{\mathbf{M}(T[i]), \mathbf{M}(T[j]), \mathbf{M}(T[k])\} \subset \mathbb{N}$, and

$$W_{\mathbf{M}(T[i])} = L \wedge W_{\mathbf{M}(T[j])} \neq L \wedge W_{\mathbf{M}(T[k])} = L. \tag{3}$$

- (b) $\mathbf{NU} = \{\mathcal{L} : (\exists \mathbf{M})[\mathcal{L} \subseteq \mathcal{I}(\mathbf{M}) \wedge (\forall L \in \mathcal{L})[\mathbf{M} \text{ does not exhibit a U-shape on any text for } L]]\}$.

Clearly, for all \mathcal{I} as above, $\mathbf{NU} \subseteq \mathcal{I}$.

In some parts of the literature (e.g., Carlucci et al. 2007b), an iterative learner is given a different formulation from that of Definition 1(d). Specifically, in (Carlucci et al. 2007b), such a learner is defined as: a pair consisting of a partial computable function of type $\mathbb{N}_? \times \mathbb{N}_\# \rightarrow \mathbb{N}_?$ and an initial conjecture. The two arguments of the partial computable function represent, respectively, the most recent conjecture (where, initially, this is the second element of the pair) and the most recent input element. The *equivalence* of the formulation of Definition 1(d) and that of (Carlucci et al. 2007b) is given by Propositions 1 and 2 below. Furthermore, these propositions provide algorithmic translations between the two formulations that *preserve non-U-shapedness*.

We prefer our formulation of Definition 1(d) to that of (Carlucci et al. 2007b) since it makes the partial function type of the learner match that of other models, e.g., **Ex**. We further note an interesting similarity between iterative learners of either kind and an *automaton* with a potentially infinite set of states, corresponding to the learner’s conjectures. It was thinking of iterative learners in this way, and the Myhill-Nerode Theorem (Davis et al. 1994), that led us to formulate iterative learners as in Definition 1(d).

Definition 4 Suppose that \mathcal{M} is a partial computable function of type $\mathbb{N}_? \times \mathbb{N}_\# \rightarrow \mathbb{N}_?$, and that $p_0 \in \mathbb{N}$ is fixed. Further suppose that L is fixed, and that T is a text for L . For all i , let $p_{i+1} = \mathcal{M}(p_i, T(i))$. Then, (a) and (b) below.

- (a) (Wiehagen 1976)⁹ (\mathcal{M}, p_0) **It-identifies** L from $T \Leftrightarrow$ (i) and (ii) below.
 - (i) For all i , $\mathcal{M}(p_i, T(i)) \downarrow$.
 - (ii) There exists i such that $(\forall j \geq i)[p_j = p_i \in \mathbb{N}]$ and $W_{p_i} = L$.
- (b) (\mathcal{M}, p_0) exhibits a *U-shape* on $T \Leftrightarrow$ there exist i, j , and k such that $i < j < k$, $\{p_i, p_j, p_k\} \subset \mathbb{N}$, and

$$W_{p_i} = L \wedge W_{p_j} \neq L \wedge W_{p_k} = L. \tag{4}$$

Proposition 1 For all \mathbf{M} , there exists a partial computable function \mathcal{M} of type $\mathbb{N}_? \times \mathbb{N}_\# \rightarrow \mathbb{N}_?$ and $p_0 \in \mathbb{N}$ satisfying the following. Let $L \in \mathbf{It}(\mathbf{M})$ be fixed, and let T be a text for L . Then, (a) through (c) below.

- (a) For all i , let p_{i+1} be as in Definition 4 for \mathcal{M} , p_0 , and T . Then, for all i , if $\mathbf{M}(T[i]) \in \mathbb{N}$, then $p_i \in \mathbb{N}$ and $W_{p_i} = W_{\mathbf{M}(T[i])}$.

⁹See also (Carlucci et al. 2007b, Definition 6).

- (b) If \mathbf{M} does not exhibit a U-shape on T , then (\mathcal{M}, p_0) does not exhibit a U-shape on T .
- (c) (\mathcal{M}, p_0) **It-identifies** L from T .

Proof By 1-1 s-m-n (Rogers 1967), there exists a 1-1, computable function $f : \mathbb{N}_\#^* \rightarrow \mathbb{N}$ such that, for all σ ,

$$W_{f(\sigma)} = \begin{cases} W_{\mathbf{M}(\sigma)}, & \text{if } \mathbf{M}(\sigma) \in \mathbb{N}; \\ \emptyset, & \text{otherwise.} \end{cases} \tag{5}$$

For all σ and $x \in \mathbb{N}_\#$, let

$$\mathcal{M}(f(\sigma), x) = \begin{cases} \uparrow, & \text{if } (*) \mathbf{M}(\sigma) \uparrow \vee \mathbf{M}(\sigma \diamond x) \uparrow; \\ f(\sigma \diamond x), & \text{if } \neg(*) \wedge \mathbf{M}(\sigma \diamond x) \neq \mathbf{M}(\sigma); \\ f(\sigma), & \text{otherwise.} \end{cases} \tag{6}$$

Let $L \in \mathbf{It}(\mathbf{M})$ be fixed, and let T be a text for L . Let $p_0 = f(\lambda)$. For all i , let $p_{i+1} = \mathcal{M}(p_i, T(i))$. The remainder of the proof is to show that, for all i , (i) through (iii) below.

- (i) $\mathcal{M}(p_i, T(i)) \downarrow$.
- (ii) There exists σ such that $p_i = f(\sigma)$ and $\mathbf{M}(T[i]) = \mathbf{M}(\sigma)$.
- (iii) If $\mathbf{M}(T[i + 1]) = \mathbf{M}(T[i])$, then $p_{i+1} = p_i$.

(a) through (c) in the statement of the proposition follow easily from (i) through (iii), just above, and from the definition of f . Specifically: (a) follows from (i) and (ii), and from the definition of f ; (b) follows from (a); and (c) follows from (a) and (iii).

For the case when $i = 0$, we give only the proof that (i) holds; the proofs that (ii) and (iii) hold are similar.

To show that $\mathcal{M}(p_0, T(0)) \downarrow$: Since $p_0 = f(\lambda)$ and f is 1-1, there is exactly one σ such that $f(\sigma) = p_0$, i.e., $\sigma = \lambda$. Thus, if it were the case that $\mathcal{M}(p_0, T(0)) \uparrow$, then, by the definition of \mathbf{M} , either $\mathbf{M}(\lambda) \uparrow$ or $\mathbf{M}(\lambda \diamond T(0)) = \mathbf{M}(T(0)) \uparrow$. But, both λ and $T(0)$ are initial segments of T , a text for a language in $\mathbf{It}(\mathbf{M})$. Thus, it must be the case that both $\mathbf{M}(\lambda) \downarrow$ and $\mathbf{M}(T(0)) \downarrow$.

Now, suppose, inductively, that (i) through (iii) hold for i . Let σ be such that $p_i = f(\sigma)$ and $\mathbf{M}(T[i]) = \mathbf{M}(\sigma)$.

To show that (i) holds for $i + 1$: Since T is a text for a language in $\mathbf{It}(\mathbf{M})$, $\mathbf{M}(T[i]) \downarrow$ and $\mathbf{M}(T[i + 1]) \downarrow$. From the fact that $\mathbf{M}(T[i]) = \mathbf{M}(\sigma)$, it follows that $\mathbf{M}(T[i + 1]) = \mathbf{M}(\sigma \diamond T(i))$. Thus, $\mathbf{M}(\sigma) \downarrow$ and $\mathbf{M}(\sigma \diamond T(i)) \downarrow$, and, therefore, $\mathcal{M}(p_i, T(i)) \downarrow$.

To show that (ii) holds for $i + 1$: By reasoning from the just previous paragraph, $\mathbf{M}(\sigma) \downarrow$ and $\mathbf{M}(\sigma \diamond T(i)) \downarrow$. If $\mathbf{M}(\sigma \diamond T(i)) \neq \mathbf{M}(\sigma)$, then $p_{i+1} = f(\sigma \diamond T(i))$, and, as already mentioned, $\mathbf{M}(T[i + 1]) = \mathbf{M}(\sigma \diamond T(i))$. On the other hand, if $\mathbf{M}(\sigma \diamond T(i)) = \mathbf{M}(\sigma)$, then $p_{i+1} = f(\sigma)$, and $\mathbf{M}(T[i + 1]) = \mathbf{M}(\sigma \diamond T(i)) = \mathbf{M}(\sigma)$.

To show that (iii) holds for $i + 1$: Suppose that $\mathbf{M}(T[i + 1]) = \mathbf{M}(T[i])$. Since $\mathbf{M}(T[i]) = \mathbf{M}(\sigma)$, $\mathbf{M}(\sigma \diamond T(i)) = \mathbf{M}(\sigma)$. Thus, $p_{i+1} = f(\sigma) = p_i$. □

Proposition 2 *Let \mathcal{M} be a partial computable function of type $\mathbb{N}_? \times \mathbb{N}_\# \rightarrow \mathbb{N}_?$, and let $p_0 \in \mathbb{N}_?$ be fixed. Then, there exists \mathbf{M} satisfying the following. Suppose that L is fixed, and that T is a text for L . Further suppose that (\mathcal{M}, p_0) **It-identifies** L from T . Then, (a) through (c) below.*

- (a) For all i , let p_{i+1} be as in Definition 4 for \mathcal{M} , p_0 , and T . Then, for all i , if $p_i \in \mathbb{N}$, then $\mathbf{M}(T[i]) \in \mathbb{N}$ and $W_{\mathbf{M}(T[i])} = W_{p_i}$.
- (b) If (\mathcal{M}, p_0) does not exhibit a U-shape on T , then \mathbf{M} does not exhibit a U-shape on T .

(c) **M It-identifies** L from T .

Proof Let \mathbf{M} be such that $\mathbf{M}(\lambda) = p_0$, and, for all ϱ , and all $x \in \mathbb{N}_\#$,

$$\mathbf{M}(\varrho \diamond x) = \mathcal{M}(\mathbf{M}(\varrho), x). \tag{7}$$

It is easy to verify that \mathbf{M} has the desired properties. □

3 It = NUIt

In this section, we prove our main result (Theorem 2), namely, that U-shapes are *unnecessary* in **It**-learning. We give some discussion of the structure of the proof following some necessary results and definitions.

Definition 5, just below, introduces a notion that we call *canniness*. Intuitively, an **It**-learner that is canny does not change its mind excessively, and, therefore, is much easier to reason about.

Definition 5 For all \mathbf{M} , \mathbf{M} is *canny* \Leftrightarrow for all σ , (a) through (c) below.

- (a) $\mathbf{M}(\sigma) \downarrow \Rightarrow \mathbf{M}(\sigma) \in \mathbb{N}$, i.e., \mathbf{M} never outputs ?.
- (b) $\mathbf{M}(\sigma \diamond \#) = \mathbf{M}(\sigma)$.
- (c) For all $x \in \mathbb{N}$, if $\mathbf{M}(\sigma \diamond x) \neq \mathbf{M}(\sigma)$, then, for all $\tau \supseteq \sigma \diamond x$, $\mathbf{M}(\tau \diamond x) = \mathbf{M}(\tau)$.

Theorem 1, just below, shows that, for any $\mathcal{L} \in \mathbf{It}$, there exists a canny learner that **It**-identifies every language in \mathcal{L} . This fact is used in the proof of Theorem 2.

Theorem 1 For all $\mathcal{L} \in \mathbf{It}$, there exists \mathbf{M}' such that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M}')$ and \mathbf{M}' is canny.

The proof of Theorem 1 is a simulation argument. Given a class of languages $\mathcal{L} \in \mathbf{It}$, we fix a learner \mathbf{M} such that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M})$. Then, we construct \mathbf{M}' from \mathbf{M} so that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M}')$ and \mathbf{M}' satisfies (a) through (c) in Definition 5. \mathbf{M}' operates, in part, by simulating \mathbf{M} , but *not* necessarily on the same inputs fed to \mathbf{M}' .

The construction of \mathbf{M}' is relatively straightforward (see (9) below). The difficulty comes in showing that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M}')$. In this regard, we fix $L \in \mathcal{L}$ and T a text for L . Then, from T , we construct another text T' for L . We show that \mathbf{M}' **It**-identifies L from T , in part, by showing that the conjecture to which \mathbf{M}' converges on T is semantically equivalent to that of \mathbf{M} on T' .

Proof of Theorem 1 Let $\mathcal{L} \in \mathbf{It}$ be fixed. Let \mathbf{M} be such that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M})$. By 1-1 s-m-n (Rogers 1967), there exists a 1-1, computable function $f : \mathbb{N}_\#^* \rightarrow \mathbb{N}$ such that, for all σ ,

$$W_{f(\sigma)} = \begin{cases} W_{\mathbf{M}(\sigma \diamond \#^m)}, & \text{where } m \text{ is least such that} \\ & \mathbf{M}(\sigma \diamond \#^{m+1}) = \mathbf{M}(\sigma \diamond \#^m) \in \mathbb{N}, \\ & \text{if such an } m \text{ exists;} \\ \emptyset, & \text{otherwise.} \end{cases} \tag{8}$$

\mathbf{M}' is such that $\mathbf{M}'(\lambda) = f(\lambda)$, and, for all ϱ and σ , and all $x \in \mathbb{N}_\#$, $[\mathbf{M}'(\varrho)\uparrow \Rightarrow \mathbf{M}'(\varrho \diamond x)\uparrow]$ and $\mathbf{M}'(\varrho) = f(\sigma) \Rightarrow \mathbf{M}'(\varrho \diamond x) =$

$$\left\{ \begin{array}{ll} \uparrow, & \text{if } (*) \ x \notin \text{content}(\sigma) \cup \{\#\} \\ & \wedge [\mathbf{M}(\sigma)\uparrow \vee \mathbf{M}(\sigma \diamond \#)\uparrow \vee \mathbf{M}(\sigma \diamond \# \diamond x)\uparrow]; \\ f(\sigma \diamond \# \diamond x), & \text{if } \neg(*) \ \wedge \ x \notin \text{content}(\sigma) \cup \{\#\} \\ & \wedge [\mathbf{M}(\sigma \diamond \#) \neq \mathbf{M}(\sigma) \vee \mathbf{M}(\sigma \diamond \# \diamond x) \neq \mathbf{M}(\sigma)]; \\ f(\sigma), & \text{otherwise.} \end{array} \right. \tag{9}$$

Clearly, \mathbf{M}' satisfies (a) and (b) in Definition 5. That \mathbf{M}' satisfies (c) in Definition 5 is demonstrated by Claim 2 below. That $\mathcal{L} \subseteq \text{It}(\mathbf{M}')$ is demonstrated by Claims 11 and 13 below. \square

Claim 1 *Let T be any text. Then, (a) and (b) below.*

- (a) *For all i, j, σ , and τ , if $\mathbf{M}'(T[i]) = f(\sigma)$, $\mathbf{M}'(T[j]) = f(\tau)$, and $i \leq j$, then $\sigma \subseteq \tau$.*
- (b) *For all i and σ , if $\mathbf{M}'(T[i]) = f(\sigma)$, then $\text{content}(\sigma) \subseteq \text{content}(T[i])$.*

Proof of Claim Easily verifiable from the definition of \mathbf{M}' . \square

Claim 2 *Let T be any text. For all i , if $T(i) \in \mathbb{N}$ and $\mathbf{M}'(T[i + 1]) \neq \mathbf{M}'(T[i])$, then, for all $j > i$ such that $T(j) = T(i)$, $\mathbf{M}'(T[j + 1]) = \mathbf{M}'(T[j])$.*

Proof of Claim Let T be any text, and let i be such that $T(i) \in \mathbb{N}$ and $\mathbf{M}'(T[i + 1]) \neq \mathbf{M}'(T[i])$. If $\mathbf{M}'(T[i + 1])\uparrow$, then, clearly, $(\forall j > i)[\mathbf{M}'(T[j])\uparrow]$, and the claim is satisfied. So, suppose that $\mathbf{M}'(T[i + 1])\downarrow$. Let σ be such that $\mathbf{M}'(T[i]) = f(\sigma)$. Clearly, by the definition of \mathbf{M}' , $\mathbf{M}'(T[i + 1]) = f(\sigma \diamond \# \diamond T(i))$. Let j be such that $j > i$ and $T(j) = T(i)$. If $\mathbf{M}'(T[j])\uparrow$, then, similarly, the claim is satisfied. So, let τ be such that $\mathbf{M}'(T[j]) = f(\tau)$. By Claim 1(a), $\sigma \diamond \# \diamond T(i) \subseteq \tau$, and, thus, $T(j) = T(i) \in \text{content}(\tau)$. Clearly, then, by the definition of \mathbf{M}' , $\mathbf{M}'(T[j + 1]) = f(\tau)$. \square

Let $L \in \mathcal{L}$ be fixed, and let T be a fixed text for L .

Claim 3 *For all i , $\mathbf{M}'(T[i])\downarrow$.*

Proof of Claim It follows from Claim 1(b) that condition $(*)$ never applies as \mathbf{M}' is fed T , and, thus, for all i , $\mathbf{M}'(T[i])\downarrow$. \square

For all i , let σ_i be such that

$$\mathbf{M}'(T[i]) = f(\sigma_i). \tag{10}$$

By Claim 3, such σ_i exist. Let $k_0 = 0$, and let

$$\{k_1 < k_2 < \dots\} = \{k > 0 : T(k - 1) \notin \text{content}(\sigma_{k-1}) \cup \{\#\}\}. \tag{11}$$

Let $\eta \in \mathbb{N} \cup \{\omega\}$ be the order type (Rogers 1967; Sierpinski 1965; Kuratowski and Mostowski 1967) of (11). Thus, η is equal to the largest subscript occurring on the left hand side of (11), if (11) is finite; $\eta = \omega$, if (11) is infinite. Let $k_{1+\eta} = \omega$. (Recall: $1 + \omega = \omega$.)

Claim 4 $(\forall i < 1 + \eta)(\forall \ell)[k_i \leq \ell < k_{i+1} \Rightarrow \sigma_\ell = \sigma_{k_i}]$.

Proof of Claim By way of contradiction, suppose that $i < 1 + \eta$, and let ℓ be *least* such that $k_i \leq \ell < k_{i+1}$ and $\sigma_\ell \neq \sigma_{k_i}$. Clearly, $\ell > k_i$. Furthermore, by the definition of \mathbf{M}' , it must be the case that $T(\ell - 1) \notin \text{content}(\sigma_{\ell-1}) \cup \{\#\}$. If $k_{i+1} < \omega$, then $k_i - 1 < \ell - 1 < k_{i+1} - 1$, which contradicts the choice of k_{i+1} . On the other hand, if $k_{i+1} = \omega$, then $k_\eta - 1 = k_i - 1 < \ell - 1$, which contradicts the choice of η . \square

Let T' be such that, for all i ,

$$T'(2i) = \#; \tag{12}$$

$$T'(2i + 1) = \begin{cases} T(k_{i+1} - 1), & \text{if } i < \eta; \\ \#, & \text{otherwise.} \end{cases} \tag{13}$$

Claim 5 T' is a text for L .

Proof of Claim Clearly, $\text{content}(T') \subseteq \text{content}(T)$. So, suppose, by way of contradiction, that $\text{content}(T') \subset \text{content}(T)$. Let ℓ be *least* such that $T(\ell) \notin \text{content}(T') \cup \{\#\}$. Since ℓ is least such, clearly, $T(\ell) \notin \text{content}(T[\ell])$, and, by Claim 1(b), $T(\ell) \notin \text{content}(\sigma_\ell)$. Thus, there must exist $i < \eta$ such that $k_{i+1} - 1 = \ell$. But then, clearly, $T'(2i + 1) = T(k_{i+1} - 1) = T(\ell)$ —a contradiction. \square

Claim 6 For all i , $\mathbf{M}'(T'[i]) \downarrow$.

Proof of Claim The reasoning is, essentially, the same as that of Claim 3. \square

For all i , let σ'_i be such that

$$\mathbf{M}'(T'[i]) = f(\sigma'_i). \tag{14}$$

By Claim 6, such σ'_i exist.

Claim 7 For all i , $\sigma'_{2i+1} = \sigma'_{2i}$.

Proof of Claim Immediate by the definition of \mathbf{M}' and the fact that, for all i , $T'(2i) = \#$. \square

Claim 8

- (a) $(\forall i < 1 + \eta)[\sigma'_{2i} = \sigma_{k_i}]$.
- (b) $(\forall i < \eta)[T'(2i + 1) \notin \text{content}(\sigma'_{2i+1}) \cup \{\#\}]$.

Proof of Claim The proof is by simultaneous induction. Clearly, (a) holds in the case when $i = 0$, since $\sigma_{k_0} = \sigma_0 = \lambda$. So, suppose, inductively, that (a) holds for i , i.e., $\sigma'_{2i} = \sigma_{k_i}$. If $i < \eta$, then to see that (b) holds for i :

$$\begin{aligned} T'(2i + 1) &= T(k_{i+1} - 1) && \{\text{by the definition of } T'\} \\ &\notin \text{content}(\sigma_{k_{i+1}-1}) \cup \{\#\} && \{\text{by the choice of } k_{i+1}\} \\ &= \text{content}(\sigma_{k_i}) \cup \{\#\} && \{\text{by Claim 4}\} \\ &= \text{content}(\sigma'_{2i}) \cup \{\#\} && \{\text{by (a) for } i\} \\ &= \text{content}(\sigma'_{2i+1}) \cup \{\#\} && \{\text{by Claim 7}\}. \end{aligned}$$

If $i + 1 < 1 + \eta$, then to see that (a) holds for $i + 1$, consider the following cases.

CASE $\mathbf{M}(\sigma_{k_{i+1}-1} \diamond \#) \neq \mathbf{M}(\sigma_{k_{i+1}-1}) \vee \mathbf{M}(\sigma_{k_{i+1}-1} \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(\sigma_{k_{i+1}-1})$. By Claim 4, this case is equivalent to

$$\mathbf{M}(\sigma_{k_i} \diamond \#) \neq \mathbf{M}(\sigma_{k_i}) \vee \mathbf{M}(\sigma_{k_i} \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(\sigma_{k_i}); \tag{15}$$

by (a) for i and the definition of T' , it is equivalent to

$$\mathbf{M}(\sigma'_{2i} \diamond \#) \neq \mathbf{M}(\sigma'_{2i}) \vee \mathbf{M}(\sigma'_{2i} \diamond \# \diamond T'(2i + 1)) \neq \mathbf{M}(\sigma'_{2i}); \tag{16}$$

and by Claim 7, it is equivalent to

$$\mathbf{M}(\sigma'_{2i+1} \diamond \#) \neq \mathbf{M}(\sigma'_{2i+1}) \vee \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) \neq \mathbf{M}(\sigma'_{2i+1}). \tag{17}$$

Thus,

$$\begin{aligned} \sigma'_{2i+2} &= \sigma'_{2i+1} \diamond \# \diamond T'(2i + 1) && \{\text{by the definition of } \mathbf{M}', \text{ (b) for } i, \\ & && \text{and (17)}\} \\ &= \sigma'_{2i} \diamond \# \diamond T'(2i + 1) && \{\text{by Claim 7}\} \\ &= \sigma_{k_i} \diamond \# \diamond T(k_{i+1} - 1) && \{\text{by (a) for } i \text{ and the definition of } T'\} \\ &= \sigma_{k_{i+1}-1} \diamond \# \diamond T(k_{i+1} - 1) && \{\text{by Claim 4}\} \\ &= \sigma_{k_{i+1}} && \{\text{by the def. of } \mathbf{M}' \text{ and the case.}\} \end{aligned}$$

CASE $\mathbf{M}(\sigma_{k_{i+1}-1} \diamond \#) = \mathbf{M}(\sigma_{k_{i+1}-1}) \wedge \mathbf{M}(\sigma_{k_{i+1}-1} \diamond \# \diamond T(k_{i+1} - 1)) = \mathbf{M}(\sigma_{k_{i+1}-1})$. By Claim 4, this case is equivalent to

$$\mathbf{M}(\sigma_{k_i} \diamond \#) = \mathbf{M}(\sigma_{k_i}) \wedge \mathbf{M}(\sigma_{k_i} \diamond \# \diamond T(k_{i+1} - 1)) = \mathbf{M}(\sigma_{k_i}); \tag{18}$$

by the (a) for i and the definition of T' , it is equivalent to

$$\mathbf{M}(\sigma'_{2i} \diamond \#) = \mathbf{M}(\sigma'_{2i}) \wedge \mathbf{M}(\sigma'_{2i} \diamond \# \diamond T'(2i + 1)) = \mathbf{M}(\sigma'_{2i}); \tag{19}$$

and by Claim 7, it is equivalent to

$$\mathbf{M}(\sigma'_{2i+1} \diamond \#) = \mathbf{M}(\sigma'_{2i+1}) \wedge \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) = \mathbf{M}(\sigma'_{2i+1}). \tag{20}$$

Thus,

$$\begin{aligned} \sigma'_{2i+2} &= \sigma'_{2i+1} && \{\text{by the definition of } \mathbf{M}' \text{ and (20)}\} \\ &= \sigma'_{2i} && \{\text{by Claim 7}\} \\ &= \sigma_{k_i} && \{\text{by (a) for } i\} \\ &= \sigma_{k_{i+1}-1} && \{\text{by Claim 4}\} \\ &= \sigma_{k_{i+1}} && \{\text{by the definition of } \mathbf{M}' \text{ and the case.}\} \quad \square \end{aligned}$$

Claim 9 For all $i < 1 + \eta$, $\mathbf{M}(T'[2i]) = \mathbf{M}(\sigma'_{2i})$.

Proof of Claim Clearly, the claim holds in the case when $i = 0$. So, let i be such that $i + 1 < 1 + \eta$ and suppose, inductively, that $\mathbf{M}(T'[2i]) = \mathbf{M}(\sigma'_{2i})$. Consider the following cases.

CASE $\mathbf{M}(\sigma'_{2i+1} \diamond \#) \neq \mathbf{M}(\sigma'_{2i+1}) \vee \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) \neq \mathbf{M}(\sigma'_{2i+1})$. Then,

$$\begin{aligned} & \mathbf{M}(T'[2i + 2]) \\ &= \mathbf{M}(T'[2i] \diamond \# \diamond T'(2i + 1)) \quad \{\text{by the definition of } T'\} \\ &= \mathbf{M}(\sigma'_{2i} \diamond \# \diamond T'(2i + 1)) \quad \{\text{by the induction hypothesis}\} \\ &= \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) \quad \{\text{by Claim 7}\} \\ &= \mathbf{M}(\sigma'_{2i+2}) \quad \{\text{by the definition of } \mathbf{M}', \\ & \quad \text{Claim 8(b), and the case}\}. \end{aligned}$$

CASE $\mathbf{M}(\sigma'_{2i+1} \diamond \#) = \mathbf{M}(\sigma'_{2i+1}) \wedge \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) = \mathbf{M}(\sigma'_{2i+1})$. Then,

$$\begin{aligned} & \mathbf{M}(T'[2i + 2]) \\ &= \mathbf{M}(T'[2i] \diamond \# \diamond T'(2i + 1)) \quad \{\text{by the definition of } T'\} \\ &= \mathbf{M}(\sigma'_{2i} \diamond \# \diamond T'(2i + 1)) \quad \{\text{by the induction hypothesis}\} \\ &= \mathbf{M}(\sigma'_{2i+1} \diamond \# \diamond T'(2i + 1)) \quad \{\text{by Claim 7}\} \\ &= \mathbf{M}(\sigma'_{2i+1}) \quad \{\text{by the case}\} \\ &= \mathbf{M}(\sigma'_{2i+2}) \quad \{\text{by the def. of } \mathbf{M}' \text{ and the case}\}. \end{aligned}$$

□

Claim 10 *If $\eta < \omega$, then, for all m ,*

$$\mathbf{M}(\sigma_{k_\eta} \diamond \#^{m+1}) \neq \mathbf{M}(\sigma_{k_\eta} \diamond \#^m) \Leftrightarrow \mathbf{M}(T'[2\eta + m + 1]) \neq \mathbf{M}(T'[2\eta + m]). \quad (21)$$

Proof of Claim Suppose that $\eta < \omega$. Then, for all m ,

$$\begin{aligned} & \mathbf{M}(\sigma_{k_\eta} \diamond \#^{m+1}) \neq \mathbf{M}(\sigma_{k_\eta} \diamond \#^m) \\ & \Leftrightarrow \mathbf{M}(\sigma'_{2\eta} \diamond \#^{m+1}) \neq \mathbf{M}(\sigma'_{2\eta} \diamond \#^m) \quad \{\text{by Claim 8(a)}\} \\ & \Leftrightarrow \mathbf{M}(T'[2\eta] \diamond \#^{m+1}) \neq \mathbf{M}(T'[2\eta] \diamond \#^m) \quad \{\text{by Claim 9}\} \\ & \Leftrightarrow \mathbf{M}(T'[2\eta + m + 1]) \neq \mathbf{M}(T'[2\eta + m]) \quad \{\text{by the definition of } T'\}. \end{aligned}$$

□

Claim 11 *If $\eta < \omega$, then \mathbf{M}' It-identifies L from T .*

Proof of Claim Suppose that $\eta < \omega$. By Claim 4, for all $\ell \geq k_\eta$, $\sigma_\ell = \sigma_{k_\eta}$.

Let m be least such that \mathbf{M} converges to $\mathbf{M}(T'[2\eta + m])$ on T' . Thus, $\mathbf{M}(T'[2\eta + m]) \in \mathbb{N}$ and $W_{\mathbf{M}(T'[2\eta + m])} = L$. By Claim 10, m is least such that $\mathbf{M}(\sigma_{k_\eta} \diamond \#^{m+1}) = \mathbf{M}(\sigma_{k_\eta} \diamond \#^m)$. Thus, $W_{f(\sigma_{k_\eta})} = W_{\mathbf{M}(\sigma_{k_\eta} \diamond \#^m)}$. Furthermore,

$$\begin{aligned} W_{\mathbf{M}(\sigma_{k_\eta} \diamond \#^m)} &= W_{\mathbf{M}(\sigma'_{2\eta} \diamond \#^m)} \quad \{\text{by Claim 8(a)}\} \\ &= W_{\mathbf{M}(T'[2\eta] \diamond \#^m)} \quad \{\text{by Claim 9}\} \\ &= W_{\mathbf{M}(T'[2\eta + m])} \quad \{\text{by the definition of } T'\} \\ &= L \quad \{\text{by the choice of } m\}. \end{aligned}$$

□

Claim 12 *For all $i < \eta$,*

$$\sigma_{k_{i+1}} \neq \sigma_{k_i} \Leftrightarrow [\mathbf{M}(T'[2i + 1]) \neq \mathbf{M}(T'[2i]) \vee \mathbf{M}(T'[2i + 2]) \neq \mathbf{M}(T'[2i])]. \quad (22)$$

Proof of Claim For all $i < \eta$,

$$\begin{aligned}
 &\sigma_{k_{i+1}} \neq \sigma_{k_i} \\
 \Leftrightarrow &\sigma_{k_{i+1}} \neq \sigma_{k_{i+1}-1} && \text{\{by Claim 4\}} \\
 \Leftrightarrow &\mathbf{M}(\sigma_{k_{i+1}-1} \diamond \#) \neq \mathbf{M}(\sigma_{k_{i+1}-1}) \\
 &\quad \vee \mathbf{M}(\sigma_{k_{i+1}-1} \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(\sigma_{k_{i+1}-1}) && \text{\{by the def. of } \mathbf{M}' \text{ and the choice of } k_{i+1}\}} \\
 \Leftrightarrow &\mathbf{M}(\sigma_{k_i} \diamond \#) \neq \mathbf{M}(\sigma_{k_i}) \\
 &\quad \vee \mathbf{M}(\sigma_{k_i} \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(\sigma_{k_i}) && \text{\{by Claim 4\}} \\
 \Leftrightarrow &\mathbf{M}(\sigma'_{2i} \diamond \#) \neq \mathbf{M}(\sigma'_{2i}) \\
 &\quad \vee \mathbf{M}(\sigma'_{2i} \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(\sigma'_{2i}) && \text{\{by Claim 8(a)\}} \\
 \Leftrightarrow &\mathbf{M}(T'[2i] \diamond \#) \neq \mathbf{M}(T'[2i]) \\
 &\quad \vee \mathbf{M}(T'[2i] \diamond \# \diamond T(k_{i+1} - 1)) \neq \mathbf{M}(T'[2i]) && \text{\{by Claim 9\}} \\
 \Leftrightarrow &\mathbf{M}(T'[2i+1]) \neq \mathbf{M}(T'[2i]) \\
 &\quad \vee \mathbf{M}(T'[2i+2]) \neq \mathbf{M}(T'[2i]) && \text{\{by the def. of } T'\}. \quad \square
 \end{aligned}$$

Claim 13 *If $\eta = \omega$, then \mathbf{M}' It-identifies L from T .*

Proof of Claim Suppose that $\eta = \omega$. Let i be such that \mathbf{M} converges to $\mathbf{M}(T'[2i])$ on T' . Thus, $\mathbf{M}(T'[2i]) \in \mathbb{N}$ and $W_{\mathbf{M}(T'[2i])} = L$. By Claim 12, for all $j \geq i$, $\sigma_{k_{j+1}} = \sigma_{k_j}$. It then follows from Claim 4 that, for all $\ell \geq k_i$, $\sigma_\ell = \sigma_{k_i}$.

Consider the behavior of \mathbf{M}' on $T[k_{i+1}]$. By the choice of k_{i+1} , $T(k_{i+1} - 1) \notin \text{content}(\sigma_{k_{i+1}-1}) \cup \{\#\}$. Thus, since $\sigma_{k_{i+1}} = \sigma_{k_i} = \sigma_{k_{i+1}-1}$, it must be the case that

$$\begin{aligned}
 &\mathbf{M}(\sigma_{k_{i+1}-1} \diamond \#) = \mathbf{M}(\sigma_{k_{i+1}-1}) \\
 &\quad \wedge \mathbf{M}(\sigma_{k_{i+1}-1} \diamond \# \diamond T(k_{i+1} - 1)) = \mathbf{M}(\sigma_{k_{i+1}-1}). && (23)
 \end{aligned}$$

Furthermore,

$$\begin{aligned}
 W_{f(\sigma_{k_i})} &= W_{f(\sigma_{k_{i+1}-1})} && \text{\{by Claim 4\}} \\
 &= W_{\mathbf{M}(\sigma_{k_{i+1}-1})} && \text{\{by the definition of } f \text{ and (23)\}} \\
 &= W_{\mathbf{M}(\sigma_{k_i})} && \text{\{by Claim 4\}} \\
 &= W_{\mathbf{M}(\sigma'_{2i})} && \text{\{by Claim 8(a)\}} \\
 &= W_{\mathbf{M}(T'[2i])} && \text{\{by Claim 9\}} \\
 &= L && \text{\{by the choice of } i\}. \quad \square
 \end{aligned}$$

Definition 6, just below, introduces notation used in the proof of Theorem 2. In this definition, C is mnemonic for *cycle*; B is mnemonic for *branch*.

Definition 6 For all \mathbf{M} and σ , (a) through (d) below.

- (a) $C_{\mathbf{M}}(\sigma) = \{x \in \mathbb{N}_{\#} : \mathbf{M}(\sigma \diamond x) \downarrow = \mathbf{M}(\sigma) \downarrow\}$.
- (b) $B_{\mathbf{M}}(\sigma) = \{x \in \mathbb{N}_{\#} : \mathbf{M}(\sigma \diamond x) \downarrow \neq \mathbf{M}(\sigma) \downarrow\}$.
- (c) $B_{\mathbf{M}}^{\cap}(\sigma) = \bigcap_{0 \leq i \leq |\sigma|} B_{\mathbf{M}}(\sigma[i])$.
- (d) $CB_{\mathbf{M}}(\sigma) = (\bigcup_{0 \leq i < |\sigma|} C_{\mathbf{M}}(\sigma[i])) \cap B_{\mathbf{M}}(\sigma)$.

Roughly, cycling on an input element corresponds to *ignoring* that element; whereas branching on an input element corresponds to *memorizing* that element. In this sense, if $\mathbf{M}(\sigma)$ is the most recent conjecture of some learner \mathbf{M} , then the elements of $B_{\mathbf{M}}^{\cap}(\sigma)$ are

those elements of $B_M(\sigma)$ that M can be sure that it has *not* yet seen; whereas the elements of $CB_M(\sigma)$ are those elements of $B_M(\sigma)$ for which M *cannot* be so sure.

Lemmas 1 through 3, just below, are used in the proof of Theorem 2.

Lemma 1 *Suppose that M and \mathcal{L} are such that $\mathcal{L} \subseteq \text{It}(M)$ and M is canny. Suppose that L and σ are such that $L \in \mathcal{L}$ and $\text{content}(\sigma) \subseteq L$. Suppose, finally, that $L \cap B_M^\cap(\sigma) = \emptyset$ and that $L \cap CB_M(\sigma)$ is finite. Then, $W_{M(\sigma)} = L$.*

Proof (Sketch) Suppose the hypotheses. Let $A = L \cap CB_M(\sigma)$. Clearly,

$$(\forall x \in A)(\exists \varrho \subset \sigma)[x \in C_M(\varrho)]. \tag{24}$$

Furthermore, since $L \cap B_M^\cap(\sigma) = \emptyset$,

$$L - A \subseteq C_M(\sigma). \tag{25}$$

Consider a text T for L described, informally, as follows. T looks, initially, like σ with the elements of A interspersed. The elements of A are positioned in T in such a way that M does *not* make a mind-change when encountering these elements. The ϱ in (24) make this possible.¹⁰ Beyond this initial sequence resembling σ , T consists of the elements of $L - A$ and, possibly, pauses ($\#$), in any order. Clearly, by (25) and the fact that M is canny, M converges to $M(\sigma)$ on such a text T . Thus, it must be the case that $W_{M(\sigma)} = L$. \square

Lemma 2 *Suppose that M , \mathcal{L} , L , and σ are as in Lemma 1. Suppose, in addition, that L is finite. Then, for all τ such that $[\sigma \subseteq \tau \wedge \text{content}(\tau) \subseteq L]$, $W_{M(\tau)} = L$.*

Proof Suppose the hypotheses, and let τ be such that $\sigma \subseteq \tau$ and $\text{content}(\tau) \subseteq L$. Since $L \cap B_M^\cap(\sigma) = \emptyset$ and $\sigma \subseteq \tau$, clearly, $L \cap B_M^\cap(\tau) = \emptyset$. Furthermore, since L is finite, $L \cap CB_M(\tau)$ is finite. Thus, by Lemma 1, $W_{M(\tau)} = L$. \square

Lemma 3 *Suppose that M and \mathcal{L} are such that $\mathcal{L} \subseteq \text{It}(M)$. Suppose that L and σ are such that $L \in \mathcal{L}$ and $\text{content}(\sigma) \subseteq L$. Suppose, finally, that $L \cap B_M(\sigma)$ is infinite. Then, for all texts T for L , and all i , there exists $j \geq i$ such that $T(j) \in B_M(\sigma)$.*

Proof Suppose the hypotheses. By way of contradiction, let T and i be such that, for all $j \geq i$, $T(j) \notin B_M(\sigma)$. Then it must be the case that $L \cap B_M(\sigma) \subseteq \{T(0), \dots, T(i - 1)\} \cap B_M(\sigma)$. But since $L \cap B_M(\sigma)$ is infinite and $\{T(0), \dots, T(i - 1)\} \cap B_M(\sigma)$ is finite, this is a contradiction. \square

Theorem 2, just below, is our main result.

Theorem 2 It = NUIt.

¹⁰For example, suppose that $x \in A$. Let $\varrho \subset \sigma$ be such that $x \in C_M(\varrho)$. Then, x is inserted into σ after ϱ , but before the remainder of σ . Since $x \in C_M(\varrho)$, $M(\varrho \diamond x) \downarrow = M(\varrho) \downarrow$. Thus, M does *not* make a mind-change when encountering x . Of course, it could be the case that more than one such x corresponds to a single ϱ . But it is straightforward to show that this case is handled, as well.

The proof of Theorem 2 is a simulation argument. Given a class of languages $\mathcal{L} \in \mathbf{It}$, we fix a learner \mathbf{M} such that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M})$. By Theorem 1, \mathbf{M} may be assumed canny. We construct \mathbf{M}' from \mathbf{M} so that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M}')$ and \mathbf{M}' does *not* exhibit a U-shape on any text for a language in \mathcal{L} . \mathbf{M}' operates, in part, by simulating \mathbf{M} , but *not* necessarily on the same inputs fed to \mathbf{M}' .

Suppose $L \in \mathbf{It}(\mathbf{M})$. For \mathbf{M} to exhibit a U-shape on some text T for L , there must be some prefix σ of T and $w \in L$ such that

$$W_{\mathbf{M}(\sigma)} = L \wedge W_{\mathbf{M}(\sigma \diamond w)} \neq L. \tag{26}$$

Thus, for such a pair (σ, w) , we have

$$w \in W_{\mathbf{M}(\sigma)} \cap B_{\mathbf{M}(\sigma)}. \tag{27}$$

The set of all pairs, (σ, w) , as in (27), is recursively enumerable. Thus, \mathbf{M}' can, in the limit, discover the existence of each such pair. Furthermore, these pairs serve as indicators that *perhaps* \mathbf{M} is in the midst of a U-shape. By detecting these pairs, \mathbf{M}' can avoid blindly following \mathbf{M} into a U-shape.

Proof of Theorem 2 Clearly, $\mathbf{NUIt} \subseteq \mathbf{It}$. Thus, it suffices to show that $\mathbf{It} \subseteq \mathbf{NUIt}$. Let $\mathcal{L} \in \mathbf{It}$ be fixed. Let \mathbf{M} be such that $\mathcal{L} \subseteq \mathbf{It}(\mathbf{M})$. Without loss of generality, assume that \mathbf{M} is canny. Let $p_{\mathbf{M}}$ be such that

$$\varphi_{p_{\mathbf{M}}} = \mathbf{M}. \tag{28}$$

Let $e : \mathbb{N}_{\#}^* \times \mathbb{N} \rightarrow \mathbb{N}$ be a partial computable function such that, for all σ , (a) through (c) below.

- (a) $\text{dom}(e(\sigma, \cdot))$ is an initial segment of \mathbb{N} .
- (b) $e(\sigma, \cdot)$ is 1-1.
- (c) $\text{rng}(e(\sigma, \cdot)) = W_{\mathbf{M}(\sigma)}$.

Clearly, such an e exists. By Kleene’s Recursion Theorem (Rogers 1967, p. 214, Problem 11-4), there exists a φ -program p_0 with the following properties. For all σ, m , and α , the elements of W_{p_0} of the form $\langle \sigma, m, \alpha, \cdot \rangle$ are chosen by the following staged construction. This construction is executed in a dove-tailing manner (Rogers 1967, p. 60), for each choice of σ, m , and α .

STAGE $s \geq 0$. If $e(\sigma, s) \downarrow$, then let $x = e(\sigma, s)$, and let

$$A = \{w : \langle \sigma, m, \alpha, w \rangle \text{ is listed in } W_{p_0} \text{ by the beginning of stage } s\} \cup \{x\}. \tag{29}$$

If *each* of (a) through (d) below is satisfied, then list $\langle \sigma, m, \alpha, x \rangle$ in W_{p_0} and proceed to stage $s + 1$; otherwise, go into an infinite loop thereby adding no more elements of the form $\langle \sigma, m, \alpha, \cdot \rangle$ to W_{p_0} .

- (a) $e(\sigma, s) \downarrow$.
- (b) $\mathbf{M}(\sigma \diamond x) \downarrow$.
- (c) $x \in C_{\mathbf{M}}(\sigma) \cup CB_{\mathbf{M}}(\sigma)$.

¹¹Clearly, $w \in L \cup \{\#\}$. However, since \mathbf{M} is canny, for all σ , $\mathbf{M}(\sigma \diamond \#) = \mathbf{M}(\sigma)$. Thus, $w \neq \#$.

$$(d) (\forall w \in A)[w \in CB_M(\sigma) \Rightarrow w \leq m] \vee (\forall \tau)[[\sigma \subset \tau \wedge \text{content}(\tau) \subseteq A \wedge |\tau| \leq |A|] \Rightarrow A \subseteq \{w : \langle \tau, 0, \lambda, w \rangle \in W_{p_0}\}].$$

By 1-1 s-m-n (Rogers 1967), there exists a 1-1, computable function $f : \mathbb{N}_\#^* \times \mathbb{N} \times \mathbb{N}_\#^* \rightarrow \mathbb{N}$ such that, for all σ, m , and α ,

$$W_{f(\sigma, m, \alpha)} = \{w : \langle \sigma, m, \alpha, w \rangle \in W_{p_0}\}. \tag{30}$$

Claim 1

- (a) For all σ, m , and $\alpha, W_{f(\sigma, m, \alpha)} \subseteq W_{M(\sigma)}$.
- (b) For all σ, m, n , and α , if $m \leq n$, then $W_{f(\sigma, m, \alpha)} \subseteq W_{f(\sigma, n, \alpha)}$.
- (c) For all σ, m, α , and $\beta, W_{f(\sigma, m, \alpha)} = W_{f(\sigma, m, \beta)}$.

Proof of Claim We give only the proof of (a); the proofs of (b) and (c) are similar. Clearly, for all σ, m , and α ,

$$\{w : \langle \sigma, m, \alpha, w \rangle \in W_{p_0}\} \subseteq \text{rng}(e(\sigma, \cdot)) = W_{M(\sigma)}. \tag{31}$$

Thus, for all σ, m , and α ,

$$\begin{aligned} W_{f(\sigma, m, \alpha)} &= \{w : \langle \sigma, m, \alpha, w \rangle \in W_{p_0}\} \quad \{\text{by (30)}\} \\ &\subseteq W_{M(\sigma)} \quad \{\text{by (31)}\}. \quad \square \end{aligned}$$

Let P be such that, for all σ and m , and all $x \in \mathbb{N}_\#$, $P(\sigma, m, x) \Leftrightarrow x \neq \#$ and

$$(\exists w)[\Phi_{M(\sigma)}(w) \leq x \wedge \Phi_{p_M}(\sigma \diamond w) \leq x \wedge w \in CB_M(\sigma) \wedge m < w \leq x]. \tag{32}$$

Note that P is a computable predicate. Let M' be such that $M'(\lambda) = f(\lambda, 0, \lambda)$, and, for all ϱ, σ, m , and α , and all $x \in \mathbb{N} \cup \{\#\}$, if $M'(\varrho) \uparrow$, then $M'(\varrho \diamond x) \uparrow$; furthermore, if $M'(\varrho) = f(\sigma, m, \alpha)$, then $M'(\varrho \diamond x)$ is:

- \uparrow , if (i) $M(\sigma) \uparrow \vee M(\sigma \diamond x) \uparrow \vee M(\sigma \diamond \alpha) \uparrow \vee M(\sigma \diamond \alpha \diamond x) \uparrow$;
- $f(\sigma \diamond \alpha \diamond x, 0, \lambda)$, if (ii) $\neg(i) \wedge [x \in B_M^\Omega(\sigma) \vee [x \in CB_M(\sigma) \wedge x > m]]$;
- $f(\sigma, m, \alpha \diamond x)$, if (iii) $\neg(i) \wedge x \in CB_M(\sigma \diamond \alpha) \wedge x \leq m$;
- $f(\sigma, x, \lambda)$, if (iv) $\neg(i) \wedge x \in C_M(\sigma \diamond \alpha) \wedge P(\sigma, m, x) \wedge \alpha = \lambda$;
- $f(\sigma \diamond \alpha, 0, \lambda)$, if (v) $\neg(i) \wedge x \in C_M(\sigma \diamond \alpha) \wedge P(\sigma, m, x) \wedge \alpha \neq \lambda$;
- $f(\sigma, m, \alpha)$, if (vi) $\neg(i) \wedge x \in C_M(\sigma \diamond \alpha) \wedge \neg P(\sigma, m, x)$.

Let $L \in \mathcal{L}$ be fixed, and let T be a fixed text for L .

Claim 2 For all $i, M'(T[i]) \downarrow$.

Proof of Claim Clearly, for all i, σ, m , and α , if $M'(T[i]) = f(\sigma, m, \alpha)$, then $\text{content}(\sigma) \cup \text{content}(\alpha) \subseteq \text{content}(T[i]) \subseteq L$. It follows that condition (i) never applies as M' is fed T , and, thus, for all $i, M'(T[i]) \downarrow$. \square

For all i , let σ_i, m_i , and α_i be such that

$$M'(T[i]) = f(\sigma_i, m_i, \alpha_i). \tag{33}$$

By Claim 2, such σ_i , m_i , and α_i exist.

Claim 3 For all i , (a) through (e) below.

- (a) $\sigma_i \diamond \alpha_i \subseteq \sigma_{i+1} \diamond \alpha_{i+1} \subseteq \sigma_i \diamond \alpha_i \diamond T(i)$.
- (b) If $T(i) \in B_M(\sigma_i \diamond \alpha_i)$, then $\sigma_{i+1} \diamond \alpha_{i+1} = \sigma_i \diamond \alpha_i \diamond T(i)$.
- (c) If $T(i) \in B_M^\cap(\sigma_i)$, then $\sigma_{i+1} = \sigma_i \diamond \alpha_i \diamond T(i)$.
- (d) If $\sigma_i = \sigma_{i+1}$, then $m_i \leq m_{i+1}$.
- (e) $\mathbf{M}(T[i])\downarrow = \mathbf{M}(\sigma_i \diamond \alpha_i)\downarrow$.

Proof of Claim (a) through (d) are easily verifiable from the definition of \mathbf{M}' . (e) follows from (a) and (b). □

Claim 4 There exists i such that, for all $j \geq i$, condition (vi) applies in calculating $\mathbf{M}'(T[j + 1])$.

Proof of Claim Suppose, by way of contradiction, that one or more of conditions (i) through (v) applies infinitely often as \mathbf{M}' is fed T . By Claim 2, condition (i) never applies as \mathbf{M}' is fed T . Also, note that, for all i , if condition (v) applies in calculating $\mathbf{M}'(T[i + 1])$, then $\alpha_i \neq \lambda$ and $\alpha_{i+1} = \lambda$. Furthermore, for all i , if $\alpha_i = \lambda$ and $\alpha_{i+1} \neq \lambda$, then condition (iii) applies in calculating $\mathbf{M}'(T[i + 1])$. Thus, if condition (v) applies infinitely often, then it must also be the case that condition (iii) applies infinitely often. Therefore, it suffices to consider the following cases.

CASE condition (iii) applies infinitely often. Then, for infinitely many i , $T(i) \in B_M(\sigma_i \diamond \alpha_i)$. Furthermore, by Claim 3(e), for infinitely many i , $T(i) \in B_M(T[i])$. Thus, \mathbf{M} does not converge on T —a contradiction.

CASE condition (ii) applies infinitely often, but condition (iii) applies only finitely often. Let i be such that, for all $j \geq i$, condition (iii) does not apply in calculating $\mathbf{M}'(T[j + 1])$. Let j be such that $j \geq i$ and $\alpha_j = \lambda$. Since condition (ii) applies infinitely often, such a j must exist.

Clearly, by the definition of \mathbf{M}' ,

$$(\forall k \geq j)[\alpha_k = \lambda]. \tag{34}$$

Since condition (ii) applies infinitely often, for infinitely many $k \geq j$,

$$\begin{aligned} T(k) &\in B_M(\sigma_k) \\ &= B_M(\sigma_k \diamond \alpha_k) \quad \{\text{by (34)}\} \\ &= B_M(T[k]) \quad \{\text{by Claim 3(e)}\}. \end{aligned}$$

Thus, \mathbf{M} does not converge on T —a contradiction.

CASE condition (iv) applies infinitely often, but conditions (ii) and (iii) apply only finitely often. Let i be such that, for all $j \geq i$, neither condition (ii) nor (iii) applies in calculating $\mathbf{M}'(T[j + 1])$. Let j be such that $j \geq i$ and $\alpha_j = \lambda$. Since condition (iv) applies infinitely often, such a j must exist.

Clearly, by the definition of \mathbf{M}' ,

$$(\forall k \geq j)[\sigma_k = \sigma_j \wedge \alpha_k = \lambda]. \tag{35}$$

Furthermore, for all $k \geq j$,

$$\begin{aligned} \mathbf{M}(T[k]) &= \mathbf{M}(\sigma_k \diamond \alpha_k) \quad \{\text{by Claim 3(e)}\} \\ &= \mathbf{M}(\sigma_j) \quad \{\text{by (35)}\}. \end{aligned}$$

Thus, \mathbf{M} converges to $\mathbf{M}(\sigma_j)$ on T , and, therefore, $W_{\mathbf{M}(\sigma_j)} = L$. Since condition (iv) applies infinitely often, it must be the case that $W_{\mathbf{M}(\sigma_j)} \cap CB_{\mathbf{M}(\sigma_j)}$ is infinite. Thus, $L \cap CB_{\mathbf{M}(\sigma_j)}$ is infinite. By Lemma 3, there exists $k \geq j$ such that $T(k) \in B_{\mathbf{M}(\sigma_j)}$. Thus, there exists $k \geq j$ such that $T(k) \in B_{\mathbf{M}(\sigma_k \diamond \alpha_k)}$. But then, clearly, condition (ii) or (iii) applies in calculating $\mathbf{M}'(T[k + 1])$ —a contradiction. □

Henceforth, let k_1 be *least* such that

$$(\forall i \geq k_1)[\text{condition (vi) applies in calculating } \mathbf{M}'(T[i + 1])]. \tag{36}$$

By Claim 4, such a k_1 exists.

Claim 5 For all $i \geq k_1$, (a) through (g) below.

- (a) $\sigma_i = \sigma_{k_1}$.
- (b) $m_i = m_{k_1}$.
- (c) $\alpha_i = \alpha_{k_1}$.
- (d) $T(i) \in C_{\mathbf{M}(\sigma_{k_1})} \cup CB_{\mathbf{M}(\sigma_{k_1})}$.
- (e) $T(i) \in CB_{\mathbf{M}(\sigma_{k_1})} \Rightarrow T(i) \leq m_{k_1}$.
- (f) $\neg P(\sigma_{k_1}, m_{k_1}, T(i))$.
- (g) $\mathbf{M}'(T[i]) = \mathbf{M}'(T[k_1])$.

Proof of Claim (a) through (f) follow from the definition of \mathbf{M}' and the choice of k_1 . (g) follows from (a) through (c). □

Claim 6 $L \cap B_{\mathbf{M}}^\cap(\sigma_{k_1}) = \emptyset$.

Proof of Claim By way of contradiction, let x be such that $x \in L \cap B_{\mathbf{M}}^\cap(\sigma_{k_1})$. By Claim 5(d), there exists $i < k_1$ such that $T(i) = x$. Clearly, $x \in B_{\mathbf{M}}^\cap(\sigma_i)$. Thus, by Claim 3(c), it must be the case that $x \in \text{content}(\sigma_{k_1})$. But this contradicts the assumption that \mathbf{M} is canny. □

Henceforth, let k_0 be *least* such that

$$L \cap B_{\mathbf{M}}^\cap(\sigma_{k_0}) = \emptyset. \tag{37}$$

By Claim 6, such a k_0 exists.

Claim 7 For all $i < k_0$, $L \not\subseteq W_{\mathbf{M}'(T[i])}$.

Proof of Claim Let i be such that $i < k_0$. By the choice of k_0 , there exists x such that $x \in L \cap B_{\mathbf{M}}^\cap(\sigma_i)$. Since $x \in B_{\mathbf{M}}^\cap(\sigma_i)$, clearly, by the definition of f , $x \notin W_{\mathbf{M}'(T[i])}$. □

Claim 8 If L is finite, then, for all σ' such that $[\sigma_{k_0} \subseteq \sigma' \wedge \text{content}(\sigma') \subseteq L]$, (a) and (b) below.

- (a) $W_{\mathbf{M}(\sigma')} = L$.
- (b) $W_{\mathbf{M}(\sigma')} \cap B_{\mathbf{M}}^{\cap}(\sigma') = \emptyset$.

Proof of Claim (a) is immediate by Lemma 2. (b) follows from (a) and the choice of k_0 . \square

Let Q be such that, for all σ' , $Q(\sigma') \Leftrightarrow$ for all τ ,

$$[\sigma' \subset \tau \wedge \text{content}(\tau) \subseteq W_{\mathbf{M}(\sigma')} \wedge |\tau| \leq |W_{\mathbf{M}(\sigma')}|] \Rightarrow W_{\mathbf{M}(\sigma')} \subseteq W_{f(\tau,0,\lambda)}. \tag{38}$$

Claim 9 *If L is finite, then, for all σ' such that $[\sigma_{k_0} \subseteq \sigma' \wedge \text{content}(\sigma') \subseteq L \wedge Q(\sigma')]$, $L \subseteq W_{f(\sigma',0,\lambda)}$.*

Proof of Claim Suppose that L is finite. Let σ' be such that $\sigma_{k_0} \subseteq \sigma'$, $\text{content}(\sigma') \subseteq L$, and $Q(\sigma')$. By Claim 8(a), $W_{\mathbf{M}(\sigma')} = L$. Consider the elements of the form $\langle \sigma', 0, \lambda, \cdot \rangle$ as they are listed in W_{p_0} . Clearly, if it can be shown that, for each stage s in which $e(\sigma', s) \downarrow$, conditions (b) through (d) in the construction of W_{p_0} are satisfied, then $L \subseteq W_{f(\sigma',0,\lambda)}$.

Let s be such that $e(\sigma', s) \downarrow$. Let x and A be as in stage s as the elements of the form $\langle \sigma', 0, \lambda, \cdot \rangle$ are listed in W_{p_0} . Since $x \in W_{\mathbf{M}(\sigma')} = L$, clearly, $\mathbf{M}(\sigma' \diamond x) \downarrow$. Furthermore, by Claim 8(b), $W_{\mathbf{M}(\sigma')} \cap B_{\mathbf{M}}^{\cap}(\sigma') = \emptyset$. Thus, since $x \in W_{\mathbf{M}(\sigma')}$, $x \in C_{\mathbf{M}}(\sigma') \cup CB_{\mathbf{M}}(\sigma')$. Finally, let τ be such that

$$\sigma' \subset \tau \wedge \text{content}(\tau) \subseteq A \wedge |\tau| \leq |A|. \tag{39}$$

Then, since $A \subseteq W_{\mathbf{M}(\sigma')}$,

$$\sigma' \subset \tau \wedge \text{content}(\tau) \subseteq W_{\mathbf{M}(\sigma')} \wedge |\tau| \leq |W_{\mathbf{M}(\sigma')}|. \tag{40}$$

Thus,

$$\begin{aligned} A &\subseteq W_{\mathbf{M}(\sigma')} \\ &\subseteq W_{f(\tau,0,\lambda)} && \{\text{by (40) and } Q(\sigma')\} \\ &= \{w : \langle \tau, 0, \lambda, w \rangle \in W_{p_0}\} && \{\text{by (30)}\}. \end{aligned} \quad \square$$

Claim 10 *If L is finite, then, for all σ' such that $[\sigma_{k_0} \subseteq \sigma' \wedge \text{content}(\sigma') \subseteq L]$, $Q(\sigma')$.*

Proof of Claim Suppose that L is finite. Let σ' be such that $\sigma_{k_0} \subseteq \sigma'$ and $\text{content}(\sigma') \subseteq L$. By Claim 8(a), $W_{\mathbf{M}(\sigma')} = L$. Thus, if $|\sigma'| \geq |L|$, then $Q(\sigma')$ holds vacuously. So, suppose, inductively, that

$$(\forall \sigma'')[[\sigma_{k_0} \subseteq \sigma'' \wedge \text{content}(\sigma'') \subseteq L \wedge |\sigma'| < |\sigma''] \Rightarrow Q(\sigma'')]. \tag{41}$$

Let τ be such that $\sigma' \subset \tau$ and $\text{content}(\tau) \subseteq W_{\mathbf{M}(\sigma')}$. Clearly, $\sigma_{k_0} \subseteq \tau$, $\text{content}(\tau) \subseteq L$, and $|\sigma'| < |\tau|$. Thus, by (41), $Q(\tau)$. Furthermore,

$$\begin{aligned} W_{f(\tau,0,\lambda)} &\supseteq L && \{\text{by Claim 9}\} \\ &= W_{\mathbf{M}(\sigma')} && \{\text{by Claim 8(a)}\}. \end{aligned} \quad \square$$

Claim 11 *If L is finite, then, for all σ' such that $[\sigma_{k_0} \subseteq \sigma' \wedge \text{content}(\sigma') \subseteq L]$, $L \subseteq W_{f(\sigma',0,\lambda)}$.*

Proof of Claim Immediate by Claims 9 and 10. \square

Claim 12 *If L is finite, then, for all $i \geq k_0$, $W_{\mathbf{M}'(T[i])} = L$.*

Proof of Claim Suppose that L is finite, and let i be such that $i \geq k_0$. Clearly, by the definition of \mathbf{M}' , $\sigma_{k_0} \subseteq \sigma_i$. Thus,

$$\begin{aligned} L &\subseteq W_{f(\sigma_i, 0, \lambda)} && \{\text{by Claim 11}\} \\ &\subseteq W_{\mathbf{M}'(T[i])} && \{\text{by (b) and (c) of Claim 1}\} \\ &\subseteq W_{\mathbf{M}(\sigma_i)} && \{\text{by Claim 1(a)}\} \\ &= L && \{\text{by Claim 8(a)}\}. \end{aligned} \quad \square$$

Claim 13 *If L is finite, then, for all i , $W_{\mathbf{M}'(T[i])} = L \Leftrightarrow i \geq k_0$.*

Proof of Claim Immediate by Claims 7 and 12. □

Claim 14 *If L is finite, then \mathbf{M}' It-identifies L from T , and, furthermore, \mathbf{M}' does not exhibit a U -shape on T .*

Proof of Claim Immediate by Claims 5(g) and 13. □

Claim 15 *For all i such that $k_0 \leq i < k_1$, if $\sigma_i \neq \sigma_{i+1}$, then there exists $w \in (L \cup W_{\mathbf{M}(\sigma_i)}) \cap CB_{\mathbf{M}(\sigma_i)}$ such that $w > m_i$.*

Proof of Claim Let i be such that $k_0 \leq i < k_1$ and $\sigma_i \neq \sigma_{i+1}$. Clearly, one of the following cases must apply.

CASE condition (ii) applies in calculating $\mathbf{M}'(T[i + 1])$. Then, clearly, $T(i) \in L \cap CB_{\mathbf{M}(\sigma_i)}$ and $T(i) > m_i$.

CASE condition (v) applies in calculating $\mathbf{M}'(T[i + 1])$. Then, since $P(\sigma_i, m_i, T(i))$, clearly, there exists $w \in W_{\mathbf{M}(\sigma_i)} \cap CB_{\mathbf{M}(\sigma_i)}$ such that $w > m_i$. □

Claim 16 *For all i such that $k_0 \leq i < k_1$, if there exists j such that $i < j \leq k_1$ and $\sigma_i \neq \sigma_j$, then there exists $w \in (L \cup W_{\mathbf{M}(\sigma_i)}) \cap CB_{\mathbf{M}(\sigma_i)}$ such that $w > m_i$.*

Proof of Claim Let i be such that $k_0 \leq i < k_1$, and let j be least such that $i < j \leq k_1$ and $\sigma_i \neq \sigma_j$. By Claim 15, there exists $w \in (L \cup W_{\mathbf{M}(\sigma_{j-1})}) \cap CB_{\mathbf{M}(\sigma_{j-1})} = (L \cup W_{\mathbf{M}(\sigma_i)}) \cap CB_{\mathbf{M}(\sigma_i)}$ such that $w > m_{j-1}$. Furthermore, by Claim 3(d), $m_{j-1} \geq m_i$, and, thus, $w > m_i$. □

Claim 17 *If L is infinite, then, for all i and j such that $k_0 \leq i < j \leq k_1$, if $L \subseteq W_{\mathbf{M}'(T[i])}$, then $W_{\mathbf{M}'(T[i])} \subseteq W_{\mathbf{M}'(T[j])}$.*

Proof of Claim By way of contradiction, suppose that L is infinite, and let i and j be such that $k_0 \leq i < j \leq k_1$, $L \subseteq W_{\mathbf{M}'(T[i])}$, and $W_{\mathbf{M}'(T[i])} \not\subseteq W_{\mathbf{M}'(T[j])}$. By Claim 1(a), $L \subseteq W_{\mathbf{M}(\sigma_i)}$. By (b) and (c) of Claim 1, it must be the case that $\sigma_i \subset \sigma_j$. Thus, by Claim 16, there exists $w \in (L \cup W_{\mathbf{M}(\sigma_i)}) \cap CB_{\mathbf{M}(\sigma_i)} = W_{\mathbf{M}(\sigma_i)} \cap CB_{\mathbf{M}(\sigma_i)}$ such that $w > m_i$.

For all s , let x^s denote the value of x during stage s of the calculation of $f(\sigma_i, m_i, \alpha_i)$, and let A^s denote the contents of the set A during stage s of the calculation of $f(\sigma_i, m_i, \alpha_i)$. Choose s such that (a) through (f) below.

- (a) $\mathbf{M}(\sigma_i \diamond x^s) \downarrow$.
- (b) $x^s \in C_{\mathbf{M}(\sigma_i)} \cup CB_{\mathbf{M}(\sigma_i)}$.

- (c) $w \in A^s$.
- (d) $\text{content}(\sigma_j) \subseteq A^s$.
- (e) $|\sigma_j| \leq |A^s|$.
- (f) $A^s \not\subseteq W_{\mathbf{M}(T[j])}$.

Clearly, such an s exists. However, since $A^s \not\subseteq W_{\mathbf{M}(T[j])}$, by (b) and (c) of Claim 1, $A^s \not\subseteq W_{f(\sigma_j, 0, \lambda)}$. Thus, by the definition of f , it must be the case that $W_{\mathbf{M}(T[i])}$ is finite. But this contradicts $L \subseteq W_{\mathbf{M}(T[i])}$. □

Claim 18 $L \cap CB_{\mathbf{M}}(\sigma_{k_1})$ is finite.

Proof of Claim By Claim 5(e), $L \cap CB_{\mathbf{M}}(\sigma_{k_1}) \subseteq \text{content}(T[k_1]) \cup \{0, \dots, m_{k_1}\}$. □

Claim 19 $W_{\mathbf{M}(\sigma_{k_1})} = L$.

Proof of Claim Immediate by Claims 6 and 18, and by Lemma 1. □

Claim 20 If L is infinite, then $\max(L \cap CB_{\mathbf{M}}(\sigma_{k_1})) \leq m_{k_1}$.

Proof of Claim By way of contradiction, suppose that L is infinite, and let x be such that $x \in L \cap CB_{\mathbf{M}}(\sigma_{k_1})$ and $x > m_{k_1}$. Choose $i \geq k_1$ such that (a) through (c) below.

- (a) $\Phi_{\mathbf{M}(\sigma_{k_1})}(x) \leq T(i)$.
- (b) $\Phi_{P_{\mathbf{M}}}(\sigma_{k_1} \diamond x) \leq T(i)$.
- (c) $x \leq T(i)$.

By Claim 19 and the fact that L is infinite, such an i exists. Clearly, $P(\sigma_{k_1}, m_{k_1}, T(i))$. But this contradicts Claim 5(f). □

Claim 21 If L is infinite, then $W_{\mathbf{M}(T[k_1])} = L$.

Proof of Claim Follows from Claims 6, 19, and 20, and from the definition of f . □

Claim 22 If L is infinite, then there exists i such that, for all j , $W_{\mathbf{M}(T[j])} = L \Leftrightarrow j \geq i$.

Proof of Claim Immediate by Claims 7, 17, and 21. □

Claim 23 If L is infinite, then \mathbf{M}^* *It*-identifies L from T , and, furthermore, \mathbf{M}^* does not exhibit a *U*-shape on T .

Proof of Claim Immediate by Claims 5(g) and 22. □

4 Iterative-with-counter learning

This section explores a learning model that we call *iterative-with-counter learning* (**ItCtr**-learning) (Definition 7 below). In this model, each of a learner’s output conjectures can depend *only* upon the learner’s most recent conjecture, the most recent input element, *and* a *counter* indicating the number of (*not* necessarily distinct) elements so far presented to the learner. Theorems 3 and 4, together, show that **ItCtr**-learning and **SDEx**-learning are

incomparable, i.e., for each, there is a class of languages learnable by that one, but *not* the other. It follows that **ItCtr**-learning is strictly more powerful than **It**-learning, yet not as powerful as full **Ex**-learning. Finally, Problem 1 below, restates, formally, the problem that this paper leaves open.

ItCtr-learning was introduced in Definition 1(e) in Sect. 2, but is repeated here for convenience.

Definition 7

- (a) For all **M** and L , **M ItCtr**-identifies $L \Leftrightarrow \mathbf{M Ex}$ -identifies L , and, for all ϱ , σ , and τ such that $\text{content}(\varrho) \cup \text{content}(\sigma) \cup \text{content}(\tau) \subseteq L$, (i) and (ii) below.
 - (i) $\mathbf{M}(\varrho) \downarrow$.
 - (ii) $[|\varrho| = |\sigma| \wedge \mathbf{M}(\varrho) = \mathbf{M}(\sigma)] \Rightarrow \mathbf{M}(\varrho \diamond \tau) = \mathbf{M}(\sigma \diamond \tau)$.
- (b) For all **M**, $\mathbf{ItCtr}(\mathbf{M}) = \{L : \mathbf{M ItCtr}\text{-identifies } L\}$.
- (c) $\mathbf{ItCtr} = \{\mathcal{L} : (\exists \mathbf{M})[\mathcal{L} \subseteq \mathbf{ItCtr}(\mathbf{M})]\}$.

Theorem 3 Based on (Kinber and Stephan 1995, Remark on p. 238) *Let \mathcal{L} be such that*

$$\mathcal{L} = \{\{0, \dots, m\} : m \in \mathbb{N}\} \cup \{\mathbb{N} - \{0\}\}. \tag{42}$$

Then, $\mathcal{L} \in \mathbf{SDEx} - \mathbf{ItCtr}$.

Proof It is already known that $\mathcal{L} \in \mathbf{SDEx}$ (Kinber and Stephan 1995, remark on p. 238). To see that $\mathcal{L} \notin \mathbf{ItCtr}$, suppose, by way of contradiction, that **M** is such that $\mathcal{L} \subseteq \mathbf{ItCtr}(\mathbf{M})$. For all s , let σ^s be as follows.

$$\sigma^0 = \lambda. \tag{43}$$

$$\sigma^{s+1} = \begin{cases} \sigma^s \diamond \#^n \diamond (s + 1), & \text{where } n \text{ is least such that} \\ & \mathbf{M}(\sigma^s \diamond \#^n) \neq \mathbf{M}(\sigma^s), \\ & \text{if such an } n \text{ exists;} \\ \sigma^s \diamond (s + 1), & \text{otherwise.} \end{cases} \tag{44}$$

Let $T = \lim_{s \rightarrow \infty} \sigma^s$. Clearly, T is a text for $\mathbb{N} - \{0\}$. Furthermore, since **M ItCtr**-identifies $\mathbb{N} - \{0\}$, the otherwise case in (44) must hold for all but finitely many s . Thus, there must exist i such that (a) and (b) below.

- (a) $T(i) \neq \#$.
- (b) $\mathbf{M}(T[i] \diamond \#) \downarrow = \mathbf{M}(T[i]) \downarrow = \mathbf{M}(T[i + 1]) \downarrow$.

Let $L_0 = \text{content}(T[i]) \cup \{0\}$ and $L_1 = \text{content}(T[i + 1]) \cup \{0\}$. Clearly, L_0 and L_1 are in \mathcal{L} . Furthermore, since no element of $\mathbb{N} - \{0\}$ appears in T more than once, by (a) above,

$$L_0 \neq L_1. \tag{45}$$

For all $j \in \{0, 1\}$, let T'_j be such that, for all $k \neq i$,

$$T'_j(k) = \begin{cases} T(k), & \text{if } k < i; \\ 0, & \text{if } k = i + 1; \\ \#, & \text{otherwise;} \end{cases} \tag{46}$$

and, for $k = i$,

$$T'_0(k) = \#; \tag{47}$$

$$T'_1(k) = T(i). \tag{48}$$

Clearly, T'_0 is a text for L_0 and T'_1 is a text for L_1 . Note that

$$\begin{aligned} \mathbf{M}(T'_0[i + 1])\downarrow &= \mathbf{M}(T'_0[i] \diamond \#)\downarrow && \{\text{by (47)}\} \\ &= \mathbf{M}(T[i] \diamond \#)\downarrow && \{\text{by (46)}\} \\ &= \mathbf{M}(T[i] \diamond T(i))\downarrow && \{\text{by (b) above}\} \\ &= \mathbf{M}(T'_1[i] \diamond T(i))\downarrow && \{\text{by (46)}\} \\ &= \mathbf{M}(T'_1[i + 1])\downarrow && \{\text{by (48)}\}. \end{aligned}$$

Let $k \geq i + 1$ be such that, for all $j \in \{0, 1\}$ and $\ell \geq k$, $\mathbf{M}(T'_j[\ell]) = \mathbf{M}(T'_j[k]) \in \mathbb{N}$. Then, by the above, and the fact the $\mathcal{L} \subseteq \mathbf{ItCtr}(\mathbf{M})$, $\mathbf{M}(T'_0[k]) = \mathbf{M}(T'_1[k])$. But then,

$$L_0 = W_{\mathbf{M}(T'_0[k])} = W_{\mathbf{M}(T'_1[k])} = L_1, \tag{49}$$

which contradicts (45). □

Theorem 4 *Let $\langle \cdot, \cdot \rangle : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}$ be any 1-1, onto, computable function (Rogers 1967), and let \mathcal{L} be such that*

$$\mathcal{L} = \{ \{ \langle e, i \rangle : i \in \mathbb{N} \} : \varphi_e(0) \uparrow \} \cup \{ \{ \langle e, i \rangle : i \leq \varphi_e(0) \} : \varphi_e(0) \downarrow \}. \tag{50}$$

Then, $\mathcal{L} \in \mathbf{ItCtr} - \mathbf{SDEx}$.

Proof To see that $\mathcal{L} \in \mathbf{ItCtr}$, consider the following. By 1-1 s-m-n (Rogers 1967), there exists a 1-1, computable function $f : \mathbb{N} \rightarrow \mathbb{N}$ such that, for all e ,

$$W_{f(2e)} = \{ \langle e, i \rangle : i \in \mathbb{N} \}; \tag{51}$$

$$W_{f(2e+1)} = \begin{cases} \{ \langle e, i \rangle : i \leq \varphi_e(0) \}, & \text{if } \varphi_e(0) \downarrow; \\ \emptyset, & \text{otherwise.} \end{cases} \tag{52}$$

Let \mathbf{M} be such that $\mathbf{M}(\lambda) = ?$, and, for all ϱ , and all $x \in \mathbb{N}_\#$,

$$\mathbf{M}(\varrho \diamond x) = \begin{cases} f(2e), & \text{where } \mathbf{M}(\varrho) = ? \text{ and } (\exists i)[x = \langle e, i \rangle], \\ & \text{if such an } e \text{ exists;} \\ f(2e + 1), & \text{where } \mathbf{M}(\varrho) = f(2e) \text{ and } \Phi_e(0) \leq |\varrho \diamond x|, \\ & \text{if such an } e \text{ exists;} \\ \mathbf{M}(\varrho), & \text{otherwise.} \end{cases} \tag{53}$$

Clearly, $\mathcal{L} \subseteq \mathbf{ItCtr}(\mathbf{M})$.

To see that $\mathcal{L} \notin \mathbf{SDEx}$, suppose, by way of contradiction, that \mathbf{M}' is such that $\mathcal{L} \subseteq \mathbf{SDEx}(\mathbf{M}')$. By Kleene’s Recursion Theorem (Rogers 1967, p. 214, problem 11-4), there exists e such that, for all x ,

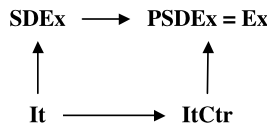
$$\varphi_e(x) = \begin{cases} m, & \text{where } m \text{ is any such that } \mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle) \in \mathbb{N} \\ & \text{and } W_{\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle)} \not\subseteq \{ \langle e, 0 \rangle, \dots, \langle e, m \rangle \}, \\ & \text{if such an } m \text{ exists;} \\ \uparrow, & \text{otherwise.} \end{cases} \tag{54}$$

Consider the following cases.

CASE $\varphi_e(0) \uparrow$. Let $L = \{\langle e, i \rangle : i \in \mathbb{N}\}$. Clearly, $L \in \mathcal{L}$. By (54), it must be the case that, for all m , either $\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle) \notin \mathbb{N}$ or $W_{\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle)} \subseteq \{\langle e, 0 \rangle, \dots, \langle e, m \rangle\}$. But then, \mathbf{M}' does *not* identify L from the text $\langle e, 0 \rangle \diamond \langle e, 1 \rangle \diamond \dots$ —a contradiction.

CASE $\varphi_e(0) \downarrow$. Let $m = \varphi_e(0)$ and $L = \{\langle e, i \rangle : i \leq m\}$. Clearly, $L \in \mathcal{L}$. By (54), it must be the case that $\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle) \in \mathbb{N}$ and $W_{\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle)} \not\subseteq \{\langle e, 0 \rangle, \dots, \langle e, m \rangle\}$. Furthermore, since \mathbf{M}' is set-driven, for all n , $\mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle \diamond \#^n) = \mathbf{M}'(\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle)$. But then, \mathbf{M}' does *not* identify L from the text $\langle e, 0 \rangle \diamond \dots \diamond \langle e, m \rangle \diamond \# \diamond \dots$ —a contradiction. □

Kinber, et al. (Kinber and Stephan 1995, Theorem 7.7 and remark on p. 238) showed that $\mathbf{It} \subset \mathbf{SDEx}$. Schäfer-Richter (1984) and Fulk (1990), independently, showed that $\mathbf{SDEx} \subset \mathbf{PSDEx}$ and that $\mathbf{PSDEx} = \mathbf{Ex}$. Clearly, $\mathbf{It} \subset \mathbf{ItCtr} \subseteq \mathbf{Ex}$. From these observations and Theorems 3 and 4, above, it follows that the *only* inclusions (represented by arrows) among \mathbf{It} , \mathbf{SDEx} , \mathbf{ItCtr} , and $\mathbf{PSDEx} = \mathbf{Ex}$ are the following.



Problem 1 *Is it the case that $\mathbf{ItCtr} = \mathbf{NUItCtr}$?*

The tools we employed to show that $\mathbf{It} = \mathbf{NUIt}$ (Theorem 2) do not seem to carry over well to the \mathbf{ItCtr} setting. For example, there does not seem to be an obvious counterpart to Theorem 1 for \mathbf{ItCtr} -learning. Of course, the difficulty of Problem 1 lies in showing that the $\mathcal{L} \in \mathbf{ItCtr} - \mathbf{It}$ are \mathbf{ItCtr} -learnable without U-shapes. At this point, we expect that a *positive* solution to Problem 1 would look more like the proof that $\mathbf{Ex} = \mathbf{NUEx}$ (Baliga et al. 2007, Theorem 20) than the proof that $\mathbf{It} = \mathbf{NUIt}$.

5 Conclusion

Herein, we solved an important problem left open in (Carlucci et al. 2007b) by showing that U-shapes are *unnecessary* in \mathbf{It} -learning (Theorem 2). In this respect, \mathbf{It} -learning likens itself to \mathbf{Ex} -learning (Baliga et al. 2007, Theorem 20), and distinguishes itself from \mathbf{Bc} -learning (Fulk et al. 1994, proof of Theorem 4) and other criteria strictly between \mathbf{Ex} and \mathbf{Bc} (Carlucci et al. 2007a). The memory limited aspect of \mathbf{It} -learning makes it more nearly applicable than \mathbf{Ex} -learning or \mathbf{Bc} -learning to modeling human learning. Thus, if one accepts \mathbf{It} -learning as an accurate model of human learning, then our result suggests that U-shapes *are* an unnecessary accident of human evolution.

We also introduced a new model of learning that we call *iterative-with-counter* learning (\mathbf{ItCtr} -learning). In this model, each of a learner’s output conjectures can depend *only* upon the learner’s most recent conjecture, the most recent input element, *and a counter* indicating the number of (*not necessarily distinct*) elements so far presented to the learner. We showed that \mathbf{ItCtr} -learning and \mathbf{SDEx} -learning are incomparable (Theorems 3 and 4). It follows that \mathbf{ItCtr} -learning is strictly more powerful than \mathbf{It} -learning, yet not as powerful as full \mathbf{Ex} -learning.

Finally, we considered the problem of whether U-shapes are necessary for **ItCtr**-learning. Unfortunately, this problem has turned out to be more difficult than we had initially anticipated. So, for now, it is left open (Problem 1).

Acknowledgements We would like to thank the anonymous referees for their helpful comments.

References

- Arikawa, S., Miyano, S., Shinohara, A., Kuhara, S., Mukouchi, Y., & Shinohara, T. (1993). A machine discovery from amino-acid-sequences by decision trees over regular patterns. *New Generation Computing*, *11*, 361–375.
- Angluin, D. (1980). Finding patterns common to a set of strings. *Journal of Computer and System Sciences*, *21*, 46–62.
- Baliga, G., Case, J., Merkle, W., Stephan, F., & Wiehagen, W. (2007). When unlearning helps. *Information and Computation*. doi:10.1016/j.ic.2007.10.005. Available online through ScienceDirect. <http://www.sciencedirect.com>.
- Blum, M. (1967). A machine independent theory of the complexity of recursive functions. *Journal of the ACM*, *14*, 322–336.
- Case, J. (1999). The power of vacillation in language learning. *SIAM Journal on Computing*, *28*(6), 1941–1969.
- Carlucci, L., Jain, S., Kinber, E., & Stephan, F. (2006). Variations on U-shaped learning. *Information and Computation*, *204*, 1264–1294.
- Carlucci, L., Case, J., Jain, S., & Stephan, F. (2007a). Non-U-shaped vacillatory and team learning. *Journal of Computer and Systems Sciences*. doi:10.1016/j.jcss.2007.06.013. Available online through ScienceDirect. <http://www.sciencedirect.com>.
- Carlucci, L., Case, J., Jain, S., & Stephan, F. (2007b). Results on memory-limited U-shaped learning. *Information and Computation*, *205*, 1551–1573.
- Case, J., & Lynes, C. (1982). Machine inductive inference and language identification. In M. Nielsen & E. Schmidt (Eds.), *Lecture notes in computer science: Vol. 140. Proceedings of the 9th international colloquium on automata, languages and programming* (pp. 107–115). Berlin: Springer.
- Case, J., & Moelius, S. E. (2007). U-shaped, iterative, and iterative-with-counter learning. In *Lecture Notes in artificial intelligence: Vol. 4539. Proceedings of the 20th annual conference on learning theory (COLT'07)* (pp. 172–186). Berlin: Springer.
- Case, J., Jain, S., Lange, S., & Zeugmann, T. (1999). Incremental concept learning for bounded data mining. *Information and Computation*, *152*, 74–110.
- Davis, M., Sigal, R., & Weyuker, E. (1994). *Computability, complexity, and languages* (2nd ed.). New York: Academic Press.
- Fulk, M. (1990). Prudence and other conditions on formal language learning. *Information and Computation*, *85*, 1–11.
- Fulk, M., Jain, S., & Osherson, D. (1994). Open problems in systems that learn. *Journal of Computer and System Sciences* *49*(3), 589–604.
- Gold, E. (1967). Language identification in the limit. *Information and Control*, *10*, 447–474.
- Jain, S. (2006). Private communication.
- Jain, S., Osherson, D., Royer, J., & Sharma, A. (1999). *Systems that learn: an introduction to learning theory* (2nd ed.). Cambridge: Cambridge University Press.
- Kinber, E., & Stephan, F. (1995). Language learning from texts: mind changes, limited memory, and monotonicity. *Information and Computation*, *123*, 224–241.
- Kuratowski, K., & Mostowski, A. (1967). *Set theory*. Amsterdam: North-Holland.
- Lange, S., & Wiehagen, R. (1991). Polynomial time inference of arbitrary pattern languages. *New Generation Computing*, *8*, 361–370.
- Lange, S., & Zeugmann, T. (1996a). Incremental learning from positive data. *Journal of Computer and System Sciences*, *53*, 88–103.
- Lange, S., & Zeugmann, T. (1996b). Set-driven and rearrangement-independent learning of recursive languages. *Mathematical Systems Theory*, *6*, 599–634.
- Marcus, G., Pinker, S., Ullman, M., Hollander, M., Rosen, T. J., & Xu, F. (1992). *Monographs of the society for research in child development: Vol. 54. Overregularization in language acquisition*. University of Chicago Press: Chicago. Includes commentary by H. Clahsen.
- Osherson, D., Stob, M., & Weinstein, S. (1986). *Systems that learn: an introduction to learning theory for cognitive and computer scientists*. Cambridge: MIT Press.

- Plunkett, K., & Marchman, V. (1991). U-shaped learning and frequency effects in a multilayered perceptron: implications for child language acquisition. *Cognition*, 38, 43–102.
- Rogers, H. (1967). *Theory of recursive functions and effective computability*. New York: McGraw Hill. Reprinted, MIT Press, 1987.
- Schäfer-Richter, G. (1984). *Über Eingabeabhängigkeit und Komplexität von Inferenzstrategien*. Rheinisch-Westfälische Technische Hochschule Aachen, Germany.
- Shimozono, S., Shinohara, A., Shinohara, T., Miyano, S., Kuhara, S., & Arikawa, S. (1994). Knowledge acquisition from amino acid sequences by machine learning system BONSAI. *Transactions of Information Processing Society of Japan*, 35, 2009–2018.
- Shinohara, T. & Arikawa, A. (1995). Pattern inference. In K.P. Jantke, & S. Lange (Eds.), *Lecture notes in artificial intelligence: Vol. 961. Algorithmic learning for knowledge-based systems* (pp. 259–291). Berlin: Springer.
- Sierpinski, W. (1965). *Cardinal and ordinal numbers* (2nd ed.). Warsaw.
- Taatgen, N. A., & Anderson, J. R. (2002). Why do children learn to say broke? A model of learning the past tense without feedback. *Cognition*, 86, 123–155.
- Wexler, K., & Culicover, P. (1980). *Formal principles of language acquisition*. Cambridge: MIT Press.
- Wiehagen, R. (1976). Limes-Erkennung rekursiver Funktionen durch spezielle Strategien. *Electronische Informationsverarbeitung und Kybernetik*, 12, 93–99.