Extensions of Embeddings in the Computably Enumerable Degrees

Jitai Zhao[1,2,⋆]

[1] State Key Lab. of Computer Science, Institute of Software, Chinese Academy of Sciences
[2] Graduate University of Chinese Academy of Sciences
jitai@ios.ac.cn

Abstract. In this paper, we study the extensions of embeddings in the computably enumerable Turing degrees. We show that for any c.e. degrees $\mathbf{x} \not\leq \mathbf{y}$, if either $\mathbf{y}$ is low or $\mathbf{x}$ is high, then there is a c.e. degree $\mathbf{a}$ such that both $\mathbf{0} < \mathbf{a} \leq \mathbf{x}$ and $\mathbf{x} \not\leq \mathbf{y} \cup \mathbf{a}$ hold.

1 Introduction

A set $A \subseteq \omega$ is called *computably enumerable* (c.e.) if and only if either $A = \emptyset$ or A is the range of a computable function.

A set A is *simple* if A is c.e. and $\bar{A}$, the complement of A, is infinite but contains no infinite c.e. set. Clearly if A is simple then A is not computable because $\bar{A}$ can not be c.e., since otherwise it contains an infinite c.e. set, i.e., $\bar{A}$.

Given sets $A, B \subseteq \omega$, we say that A is *Turing reducible* to B, if there is an oracle Turing machine Φ such that $A = \Phi(B)$ (denoted by $A \leq_{\mathrm{T}} B$). $A \equiv_{\mathrm{T}} B$ if $A \leq_{\mathrm{T}} B$ and $B \leq_{\mathrm{T}} A$. The *Turing degree* of A is defined to be $\mathbf{a} = deg(A) = \{B : B \equiv_{\mathrm{T}} A\}$.

A degree $\mathbf{a} \leq \mathbf{0}'$ is *low*, if $\mathbf{a}' = \mathbf{0}'$, and *high* if $\mathbf{a}' = \mathbf{0}''$. A set $A \leq_{\mathrm{T}} \emptyset'$ is *low* (*high*), if $deg(A)$ is low (high).

The degrees $\mathcal{D}$ form a partially ordered set under the relation $deg(A) \leq deg(B)$ if and only if $A \leq_{\mathrm{T}} B$. We write $deg(A) < deg(B)$ if $A \leq_{\mathrm{T}} B$ and $B \not\leq_{\mathrm{T}} A$.

A degree is called *computably enumerable* (c.e.), if it contains a c.e. set. Let $\mathcal{E}$ denote the class of c.e. degrees with the some ordering as that for $\mathcal{D}$. As we know $(\mathcal{D}; \leq, \cup)$ and $(\mathcal{E}; \leq, \cup)$ will form upper semi-lattices.

Given r.e. degrees $\mathbf{0} < \mathbf{b} < \mathbf{a}$, we say that $\mathbf{b}$ *cups to* $\mathbf{a}$ if there exists an r.e. degree $\mathbf{c} < \mathbf{a}$ such that $\mathbf{b} \cup \mathbf{c} = \mathbf{a}$; if no such $\mathbf{c}$ exists then $\mathbf{b}$ is an *anti-cupping witness* for $\mathbf{a}$. The r.e. degree $\mathbf{a}$ has *the anti-cupping (a.c.) property* if it has an anti-cupping witness.

Extensions and non-extensions of embeddings in the computably enumerable Turing degrees have been an extensively studied phenomena in the past decades since Shoenfield [1965] published his conjecture. The conjecture was soon proved false by the minimal pair theorem of Lachlan [1966]. However the characterization of the structure satisfying Shoenfield's conjecture has become a successful

⋆ The author is partially supported by NSFC Grant No. 60325206, and No. 60310213.

M. Agrawal et al. (Eds.): TAMC 2008, LNCS 4978, pp. 204–211, 2008.
© Springer-Verlag Berlin Heidelberg 2008

research in the local Turing degrees, leading to the final resolution of Slaman and Soare [1995] of the full characterization of the problem.

The interests in the strong extensions of embeddings in the computably enumerable degrees come from the close relationship between the problem and the decidability/undecidability of the Σ_2-fragment of the c.e. degrees. For instance, Slaman asked in [1983] the following question:

For any two c.e. degrees $\mathbf{x}$ and $\mathbf{y}$, with $\mathbf{x} \not\leq \mathbf{y}$, does there exist a c.e. degree $\mathbf{a}$, satisfying:

1. $\mathbf{0} < \mathbf{a} \leq \mathbf{x}$,
2. $\mathbf{x} \not\leq \mathbf{y} \cup \mathbf{a}$?

Intuitively, the problem wants to construct a c.e. degree $\mathbf{a}$ which should "code more information than" $\mathbf{0}$, and which cannot compute $\mathbf{x}$ even if it joins with $\mathbf{y}$.

In fact it is not always possible to find such a c.e. degree $\mathbf{a}$, which is stated in Slaman and Soare [2001]. This problem has recently been resolved negatively in progress by Barmpalias, Cooper, Li, Xia, and Yao.

In this article, we consider possible partial results on the positive side of the problem above. We consider two special cases, i.e., when $\mathbf{y}$ is low and $\mathbf{x}$ is high.

We will show the following two theorems:

Theorem 1. *For any two c.e. degrees $\boldsymbol{x}$ and $\boldsymbol{l}$, with $\boldsymbol{x} \not\leq \boldsymbol{l}$ and $\boldsymbol{l}$ low, there is a c.e. degree $\boldsymbol{a}$, satisfying:*

1. $\boldsymbol{0} < \boldsymbol{a} \leq \boldsymbol{x}$, *and*
2. $\boldsymbol{x} \not\leq \boldsymbol{l} \cup \boldsymbol{a}$.

Theorem 2. *For any two c.e. degrees $\boldsymbol{h}$ and $\boldsymbol{y}$, with $\boldsymbol{h} \not\leq \boldsymbol{y}$ and $\boldsymbol{h}$ high, there is a c.e. degree $\boldsymbol{a}$, satisfying:*

1. $\boldsymbol{0} < \boldsymbol{a} \leq \boldsymbol{h}$, *and*
2. $\boldsymbol{h} \not\leq \boldsymbol{y} \cup \boldsymbol{a}$.

Note that Theorem 2 can be directly deduced from the following theorem which appears in Miller [1981].

Theorem 3. *Every high r.e. degree $\boldsymbol{h}$ has the a.c. property via a high r.e. witness $\boldsymbol{a}$.*

We briefly describe how to show Theorem 2 by Theorem 3: for a given high r.e. degree $\mathbf{h}$, there exists a high r.e. degree $\mathbf{a}$, $\mathbf{0} < \mathbf{a} < \mathbf{h}$, and for a given r.e. $\mathbf{y}$, $\mathbf{h} \leq \mathbf{y} \cup \mathbf{a}$ implies $\mathbf{h} \leq \mathbf{y}$, i.e., $\mathbf{h} \not\leq \mathbf{y}$ implies $\mathbf{h} \not\leq \mathbf{y} \cup \mathbf{a}$. Therefore $\mathbf{a}$ is a desired r.e. degree in Theorem 2.

The rest of the paper is devoted to proving Theorem 1, our main result.

Our notations and terminology are standard and generally follow Soare [1987] and Cooper [2003]. During the course of a construction, notations such as A, Φ are used to denote the current approximations to these objects, and if we want to specify the values immediately at the end of stage s, then we denote them by A_s, $\Phi[s]$ etc. For a *computable partial functional* (c.p., or for simplicity, also a Turing functional), Φ say, the use function is denoted by the corresponding lower case letter ϕ. The value of the use function of a converging computation is the greatest number which is actually used in the computation. For a Turing functional, if a computation is not defined, then we define its use function equal to -1.

2 Proof of Theorem 1

2.1 Requirements and Strategies

Given c.e. sets $X \in \mathbf{x}$, and $L \in \mathbf{l}$, with $X \leq_{\mathrm{T}} L$ and L low, we will build a c.e. set A to satisfy the following requirements:

$$\begin{aligned} \mathcal{T} &: \ A \leq_{\mathrm{T}} X \\ \mathcal{P}_e &: \ W_e \ infinite \Rightarrow W_e \cap A \neq \emptyset \\ \mathcal{N}_e &: \ X \neq \Phi_e(L \oplus A) \end{aligned}$$

where $e \in \omega$, $\{\Phi_e : e \in \omega\}$ is an effective enumeration of all Turing reductions Φ, and W_e is the e-th c.e. set.

Let $\mathbf{a}$ be the Turing degree of A. By the $\mathcal{T}$-requirement, $\mathbf{a} \leq \mathbf{x}$, by the $\mathcal{P}$-requirements, A is simple, so it is not computable, i.e., $\mathbf{a} > \mathbf{0}$, and by the $\mathcal{N}$-requirements, $\mathbf{x} \not\leq \mathbf{l} \cup \mathbf{a} = deg(L) \cup deg(A) = deg(L \oplus A)$. Therefore the requirements are sufficient to prove the theorem.

Let $\{X_s\}_{s\in\omega}, \{L_s\}_{s\in\omega}$ be computable enumerations of X, L respectively.

During the construction, the requirements may be divided into the *positive* requirements $\mathcal{P}_e$, which attempt to put elements *into* A, and the *negative* requirements $\mathcal{N}_e$, which attempt to keep elements *out of* A, i.e., impose an A-restraint function with priority $\mathcal{N}_e$. The priority rank of the requirements is $\mathcal{N}_e < \mathcal{P}_e < \mathcal{N}_{e+1}$, for all $e \in \omega$.

First we introduce an easy method in Yates [1965] for constructing a c.e. set A which is computable in a given non-computable c.e. set B by enumerating an element x in A at some stage s only when B permits x in the sense that some element $y < x$ appears in B at the same stage s.

Proposition 1. *If $\{A_s\}_{s\in\omega}$ and $\{B_s\}_{s\in\omega}$ are computable enumerations of c.e. sets A and B respectively, such that $x \in A_{s+1} - A_s$ implies $(\exists y < x)[y \in B - B_s]$, then $A \leq_{\mathrm{T}} B$.*

Proof. To B-recursively compute whether $x \in A$, find a stage s such that $B_s \restriction x = B \restriction x$. Now $x \in A$ if and only if $x \in A_s$. □

The strategy for meeting the $\mathcal{T}$-requirement is attached onto the positive requirements. When an element x is enumerated into A, it must satisfy that $X_{s+1} \restriction x \neq X_s \restriction x$ so that $A \leq_{\mathrm{T}} X$ holds according to Proposition 1 (Soare [1987]). Note that this kind of x can always be found because X is not computable in L.

The strategy for meeting a single requirement $\mathcal{P}_e$ is the same as for Post's simple set. Intuitively, enumerate W_e until the first element $> 2e$ appears in W_e simultaneously satisfying other conditions and put it into A.

We now give a property of a low c.e. set, as found in Soare [1987].

Proposition 2. *If L is a low set then*

$$C = \{j : (\exists n \in W_j)[D_n \subseteq \bar{L}]\} \leq_{\mathrm{T}} \emptyset'. \tag{1}$$

where W_j is the j-th c.e. set, and D_n is a finite set with canonical index n.

Proof. Clearly, C is $\sum_1^L$, so $C \leq_{\mathrm{T}} L'$. If L is low then $L' \leq_{\mathrm{T}} \emptyset'$, so $C \leq_{\mathrm{T}} \emptyset'$. □

According to the Limit Lemma, let $g(e,s)$ be a computable function such that $\lim_s g(e,s)$ is the characteristic function of C.

Now we state the basic strategy for meeting a requirement $\mathcal{N}_e$, without loss of generality, let $A \subseteq 2\omega$, the even numbers, and $L \subseteq 2\omega + 1$, the odd numbers. Note that $A \oplus L \equiv_{\mathrm{T}} A \cup L$, so we use the latter from now on, i.e.,

$$\mathcal{N}_e : X \neq \Phi_e(L \cup A).$$

We follow some basic idea in Robinson [1971] of proving the Robinson Low Splitting Theorem which also can be found in Soare [1987].

Fix e and x. Intuitively, we use the lowness of L to help to "L-certify" a computation $\Phi_e((L \cup A) \upharpoonright u; x)[s]$ where $u = \phi_e(L \cup A; x)[s]$ is the use function of this computation as follows:

Let $D_n = \bar{L}_s \upharpoonright u$. Enumerate n into a c.e. V that we shall build during the construction. By the Recursion Theorem we may assume that we have in advance an index j such that $V = W_j$. Find the least $t \geq s$ such that either $D_n \cap L_t \neq \emptyset$, in which case the computation is obvious disturbed, or $g(j,t) = 1$, in which case we "L-certify" the computation and guess that it is L-correct. It may happen that we were wrong and $L \upharpoonright u \neq L_s \upharpoonright u$, but this happens at most finitely often by Proposition 2 and the Limit Lemma. Since we are really using g as an oracle to inquire whether $D_n \subseteq \bar{L}$ for the current $D_n = \bar{L}_s \upharpoonright u$, it is very important that there are no previous $m \in V_s$ unless $D_m \cap L_s \neq \emptyset$. Thus, whenever an L-certified computation first becomes A-invalid by $A_t \upharpoonright u \neq A_s \upharpoonright u$, we abandon the old c.e. set V and start with a new version of V and hence a new index j such that $W_j = V$.

This L-certification process is best formalized by transforming the function $\Phi_e(L \cup A; x)$ to a computable function $\hat{\Phi}_e(L \cup A; x)$. When we have fixed e, for notational convenience, we let

$$\hat{\Phi}_s(x) \;\leftrightarrow\; \hat{\Phi}_e(L \cup A; x)[s],$$

and

$$u_x^s \;\leftrightarrow\; \phi_e(L \cup A; x)[s].$$

If $\hat{\Phi}_{s-1}(x) \downarrow$ but $\hat{\Phi}_s(x) \uparrow$ we say that the (e,x)-computation $\hat{\Phi}_{s-1}(x) \downarrow$ becomes A-*invalid* if

$$(\exists z < u_x^{s-1})[z \in A_s - A_{s-1}]$$

and otherwise becomes L-*invalid.*

Fix e, x and s, we define $\hat{\Phi}_s(x)$ as follows: given A_t and L_t, $t \leq s$ and assume that

$$\Phi_e(L \cup A; x)[s] \downarrow= y,$$

and

$$\neg(\exists z < u_x^{s-1})[z \in (A_s \cup L_s) - (A_{s-1} \cup L_{s-1})].$$

Let $D_n = \bar{L}_s \upharpoonright u_x^s$. Enumerate n into $V_s^{e,x}$. Let v be the greatest stage less than s at which an (e, x)-computation becomes A-invalid, and $v = 0$ if no such stage exists. By the Recursion Theorem, choose j such that $W_j = \bigcup\{V_t^{e,x} : t > v\}$. Find the least $t \geq s$ such that either

$$D_n \cap L_t \neq \emptyset, \tag{2}$$

or

$$g(j, t) = 1. \tag{3}$$

If the latter holds, define $\hat{\Phi}_s(x) \downarrow= y$. Otherwise, $\hat{\Phi}_s(x) \uparrow$.

We use a strategy which is similar to the Sacks' *preserving agreement* strategy to meet a negative requirement. Here we want to construct a c.e A to meet a requirement of the form $X \neq \Phi_e(L \cup A)$. During the construction we preserve agreement between X and $\Phi_e(L \cup A)$. Sufficient preservation will guarantee that if $X = \Phi_e(L \cup A)$, then in fact $X \leq_{\mathrm{T}} L$, contrary to hypothesis.

As usual, we define the computable functions:

$$(length\ function)\ \hat{l}(e, s) = max\{x : (\forall y < x)[X_s(y) = \hat{\Phi}_s(y)]\},$$

$$(restraint\ fucntion)\ \hat{r}(e, s) = max\{u_x^s : x \leq \hat{l}(e, s)\ \&\ \hat{\Phi}_s(x) \downarrow\}.$$

We say that x *injures* $\mathcal{N}_e$ at stage $s+1$ if $x \in A_{s+1} - A_s$ and $x \leq \hat{r}(e, s)$. Define the *injury set* for $\mathcal{N}_e$,

$$(injury\ set)\ \hat{I}(e) = \{x : (\exists s)[x \in A_{s+1} - A_s\ \&\ x \leq \hat{r}(e, s)]\}.$$

The positive requirements of course are never injured.

2.2 Construction and Verification

Proof of Theorem 1.

Construction of A.

Stage $s = 0$. Set $A_0 = \emptyset$.

Stage $s + 1$. Since A_s has already been defined, we can define, for all e, the length function $\hat{l}(e, s)$ and restraint function $\hat{r}(e, s)$.

We say $\mathcal{P}_e$ *requires attention* at stage $s + 1$ if

$$W_{e,s} \cap A_s = \emptyset,$$

Then find if $\exists x$,

$$x \in W_{e,s},$$

$$x > 2e,$$

$$X_{s+1} \upharpoonright x \neq X_s \upharpoonright x,$$

and

$$(\forall i \leq e)[\hat{r}(i, s) < x].$$

Choose the least $i \leq s$ such that $\mathcal{P}_i$ requires attention, and then enumerate the least such x into A_{s+1}, and we say that $\mathcal{P}_i$ *receives attention*. Hence $W_{i,s} \cap A_{s+1} \neq \emptyset$ and $(\exists x \in A_{s+1})[X_{s+1} \upharpoonright x \neq X_s \upharpoonright x]$, so $\mathcal{P}_i$ is satisfied and never again requires attention.

If i does not exist, do nothing.

Let $A = \bigcup A_s$. This ends the construction.

To verify that the construction succeeds we must prove the following lemmas.

Lemma 1. $(\forall e)$ $[\hat{I}(e)$ *is finite*$]$. *(*$\mathcal{N}_e$ *is injured at most finitely often.)*

Proof. Note that once $\mathcal{P}_i$ receives attention, it will become satisfied and remain satisfied forever. Hence each $\mathcal{P}_i$ contributes at most one element to A, and $\mathcal{N}_e$ can be injured by $\mathcal{P}_i$ only if $i < e$. So $|\hat{I}(e)| \leq e$. □

Lemma 2. $(\forall e)$ $[X \neq \Phi_e(A \cup L)]$. *(*$\mathcal{N}_e$ *is met.)*

Proof. Assume for a contradiction that $X = \Phi_e(A \cup L)$. Then $\lim_s \hat{l}(e, s) = \infty$. By Lemma 1, choose s_1 such that $\mathcal{N}_e$ is never injured after stage s_1. We shall show that $X \leq_{\mathrm{T}} L$ contrary to hypothesis. To L-recursively compute $X(p)$ for $p \in \omega$, find some stage $s > s_1$ such that $\hat{l}(e, s) > p$ and each computation $\hat{\Phi}_s(x)$, $x \leq p$, is L-correct, namely, $L_s \upharpoonright u_x^s = L \upharpoonright u_x^s$. It follows by induction on $t \geq s$ that

$$(\forall t \geq s)[\hat{l}(e, t) > p \ \& \ \hat{r}(e, t) \geq max\{u_x^s : x \leq p\}], \tag{4}$$

and hence that for all $t \geq s$,

$$\hat{\Phi}_t(p) = \Phi_e(A \cup L; p) = X(p).$$

So X is computable in L.

To prove (4), when $t = s$, clearly it is true. Assume that it holds for t. Then by the definition of $\hat{r}(e, t)$ and $s > s_1$, for any $x \leq p$, it ensures that $(A_{t+1} \cup L_{t+1}) \upharpoonright z = (A_t \cup L_t) \upharpoonright z$ for all numbers z used in a computation $\hat{\Phi}_t(x) \downarrow= y$. Hence,

$$\hat{\Phi}_{t+1}(x) \downarrow= \hat{\Phi}_t(x) \downarrow= X_t(x).$$

So $\hat{l}(e, t+1) > p$ unless $X_{t+1}(x) \neq X_t(x)$. But if $X_t(x) \neq X_s(x)$ for some $t \geq s$, since X is c.e., the *disagreement* $\hat{\Phi}_t(x) \downarrow\neq X_t(x)$ is preserved forever, so $X(x) = X_t(x) \neq \hat{\Phi}_t(x) \downarrow= \Phi_e(A \cup L; x)$, contrary to the hypothesis $X = \Phi_e(A \cup L)$. □

Lemma 3. $(\forall e)[\lim_s \hat{r}(e, s)$ *exists and is finite*$]$.

Proof. By Lemma 1, choose s_1 such that $\mathcal{N}_e$ is never injured after stage s_1. By Lemma 2, choose $p = (\mu x)[X(x) \neq \Phi_e(A \cup L; x)]$. Choose $s_2 \geq s_1$ sufficiently large so that for all $s \geq s_2$,

$$(\forall x < p)[\hat{\Phi}_s(x) \downarrow = \Phi_e(A \cup L; x)],$$

and

$$(\forall x \leq p)[X_s(x) = X(x)].$$

Case 1. $\Phi_e(A \cup L; p) \downarrow \neq X(p)$. Choose $s_3 \geq s_2$ such that for all $s \geq s_3$, $\hat{\Phi}_s(p) \downarrow = q \neq X(p)$. Hence, for all $s \geq s_3$, $\hat{l}(e,s) = \hat{l}(e, s_3)$ and $\hat{r}(e,s) = \hat{r}(e,s_3)$.

Case 2. $\Phi_e(A \cup L; p) \uparrow$. We shall find a stage v such that for all $s \geq v$, $\hat{\Phi}_s(p) \uparrow$. Hence, for all $s \geq v$, $\hat{r}(e,s) = \hat{r}(e,v)$.

Note that if $\hat{\Phi}_s(p) \downarrow$ for any $s \geq s_2$ then $\hat{r}(e,s) \geq u_p^s$, so by induction on $t \geq s$, the computation $\hat{\Phi}_t(p) = \hat{\Phi}_s(p)$ holds as long as it remains L-valid. Let s' be the least t such that no (e,p)-computation becomes A-invalid at any stage $\geq t$. By the Recursion Theorem, choose j such that $W_j = \bigcup\{V_s^{e,p} : s \geq s'\}$. Since $\Phi_e(A \cup L; p) \uparrow$, any computation $\hat{\Phi}_s(p)$, $s \geq s'$, becomes L-invalid at some stage $t > s$, at which time $D_m \bigcap C_t \neq \emptyset$ for every $m \in V_t^{e,p}$. Hence, $\lim_s g(j,s) = 0$ by (1). Choose $v > s_2$ such that $\hat{\Phi}_v(p) \uparrow$ and $g(j,s) = 0$ for all $s \geq v$. We claim that $\hat{\Phi}_s(p) \uparrow$ for all $s \geq v$. Suppose $s > v$, $\hat{\Phi}_{s-1}(p) \uparrow$ and $\hat{\Phi}_s(p) \downarrow$. Then we enumerate $n \in V_s^{e,p}$, where $D_n = \bar{L}_s \upharpoonright u_p^s$, and we choose the least $t \geq s$ satisfying (2) or (3). But (3) could not occur by the choice of v, so (2) occurs and $\hat{\Phi}_s(p) \uparrow$. □

Lemma 4. $(\forall e)$ $[W_e$ *infinite* $\Rightarrow W_e \cap A \neq \emptyset]$. *($\mathcal{P}_e$ is met, simultaneously, $\mathcal{T}$ is met.)*

Proof. By the above lemmas, for all $i \leq e$, let

$$\hat{r}(i) = \lim_s \hat{r}(i,s)$$

and

$$\hat{R}(e) = max\{\hat{r}(i) : i \leq e\}.$$

Choose s_0 such that

$$(\forall t \geq s_0)(\forall i \leq e)[\hat{r}(e,t) = \hat{r}(e)],$$

and no $\mathcal{P}_i$, $i < e$, receives attention after stage s_0.

Now choose $s \geq s_0$, if $\exists x$,

$$x \in W_{e,s},$$

$$x > 2e,$$

$$X_{s+1} \upharpoonright x \neq X_s \upharpoonright x,$$

and

$$\hat{R}(e) < x.$$

Now either $W_{e,s} \cap A_s \neq \emptyset$ or else $\mathcal{P}_e$ receives attention at stage $s+1$, then in either case $W_{e,s} \cap A_{s+1} \neq \emptyset$, so $\mathcal{P}_e$ is met by the end of stage $s+1$. And by Proposition 1, $A \leq_{\mathrm{T}} X$ is obviously met.

It is remarkable that searching for an x with the condition $X_{s+1} \upharpoonright x \neq X_s \upharpoonright x$ does not impact the requirement $\mathcal{P}_e$. Suppose for the sake of contradiction that if W_e infinite, while $A \cap W_e = \emptyset$, then $A \subseteq \overline{W}_e$, choose an increasing c.e. sequence of elements $x_1 < x_2 < \cdots$ in W_e such that $x_1 > \hat{R}(e)$ for all stage $s \geq s_0$. Choose s_k minimal such that $s_k > s_0$ and $x_k \in W_{e,s_k}$. Now $X_{s_k} \upharpoonright x_k = X \upharpoonright x_k$ so X is computable, contrary to $X \not\leq_{\mathrm{T}} L$. □

Note that $\bar{A}$ is infinite by the clause "$x > 2e$". To see this, note that A contains at most e elements in $\{0, 1, \ldots, 2e\}$, hence $card(\bar{A} \upharpoonright (2e+1)) \geq 2e+1-e = e+1$. A is simple.

This ends the proof of Theorem 1. ■

References

1. Lachlan, A.H.: Lower bounds for pairs of recursively enumerable degrees. Proc. London Math. Soc. 16, 537–569 (1966)
2. Barmpalias, Cooper, Li, Xia, Yao, Super minimal pairs (in progress)
3. Yates, C.E.M.: Three theorems on the degrees of recursively enumerable sets. Duke Math. J.32, 461–468 (1965)
4. Miller, D.: High recursively enumerable degrees and the anti-cupping property. In: Lerman, Schmerl, and Soare, pp. 230–245 (1981)
5. Shoenfield, J.R.: Application of model theory to degrees of unsolvability. In: Addison, Henkin, and Tarski, pp. 359–363 (1965)
6. Soare, R.I.: Recursively Enumerable Sets and Degrees. Springer, Heidelberg (1987)
7. Robinson, R.W.: Interpolation and embedding in the recursively enumerable degrees. Ann. of Math. 2(93), 285–314 (1971)
8. Cooper, S.B.: Computability Theory. Chapman Hall/CRC Mathematics Series, vol. 26 (2003)
9. Slaman, T.A.: The recursively enumerable degrees as a substructure of the Δ_2^0 degrees (1983) (unpublished notes)
10. Slaman, T.A., Soare, R.I.: Algebraic aspects of the computably enumerable degrees. Proceedings of the National Academy of Science, USA 92, 617–621 (1995)
11. Slaman, T.A., Soare, R.I.: Extension of embeddings in the computably enumerable degrees. Ann. of Math (2) 154(1), 1–43 (2001)