Acta Mathematica Sinica, English Series
Sep., 2017, Vol. 33, No. 9, pp. 1184–1192
Published online: June 25, 2017
DOI: 10.1007/s10114-017-6585-5
Http://www.ActaMath.com

Acta Mathematica Sinica, English Series

Non-low$_2$-ness and Computable Lipschitz Reducibility

Yun FAN
School of Mathematics, Southeast University, Nanjing 210096*, P. R. China*
E-mail: *fanyun@seu.edu.cn*

Abstract In this paper, we prove that if a c.e. Turing degree $\mathbf{d}$ is non-low$_2$, then there are two left-c.e. reals β_0, β_1 in $\mathbf{d}$, such that, if β_0 is wtt-reducible to a left-c.e. real α, then β_1 is not computable Lipschitz (cl-) reducible to α. As a corollary, $\mathbf{d}$ contains a left-c.e. real which is not cl-reducible to any complex (wtt-complete) left-c.e. real.

Keywords Non-low$_2$, computable Lipschitz (cl) reducibility, complex

MR(2010) Subject Classification 03D25, 03D30, 03D32, 68Q30

1 Introduction

Recall that all the reals we consider are in $[0,1]$, and are identified with elements of 2^ω and with subsets of $\mathbb{N}$.

Martin–Löf randomness is a natural and robust notion of randomness in that it coincides with other methods of defining algorithmic randomness. Schnorrs theorem proved that A is Martin–Löf random if and only if for all x, $K(A \upharpoonright\upharpoonright x) = x + O(1)$, where $A \upharpoonright\upharpoonright x$ denotes the first x bits of A and K denotes prefix-free Kolmogorov complexity. Schnorrs theorem suggests a natural method of calibrating randomness of reals: $A \leq_K B$ iff for all x, $K(A \upharpoonright\upharpoonright x) \leq K(B \upharpoonright\upharpoonright x) + O(1)$. Many measures of relative randomness implying this measure have been analysed. Of interest to us here is one inspired by strong reducibilities. In [5], a strengthening of weak truth table reducibility, namely computations where the use on the oracle on argument x is $x + c$ for some constant c, i.e., for any oracle query y, $y \leq x + c$, was suggested as a possible measure of relative randomness. This reducibility has appeared in the literature with various names, e.g. *strong weak truth table* [5], *computable Lipschitz* (due to a characterization of it in terms of effective Lipschitz functions) [2, 6] and *linear* [3]. We will adopt the terminology in [2, 6] and note it as $\leq_{\mathrm{cl}}$. It is nearly obvious that $\leq_{\mathrm{cl}}$ is a measure of relative randomness. The identity bounded Turing reducibility (ibT or $\leq_{\mathrm{ibT}}$ for short) is a computable Lipschitz reduction for which the constant c is 0. It was introduced by Soare [10] in connection with applications of computability theory to differential geometry.

The cl-degrees are the equivalence classes under cl-reducibility. Notice that they either contain only random reals or only non-random reals. In [5], the structure of cl-degrees of left-c.e. reals (i.e. limits of computable increasing sequences of rationals) is neither a lower semi-lattice nor an upper semi-lattice. In [11], Yu and Ding proved that there are two left-c.e. reals which

Received December 23, 2016, revised March 13, 2017, accepted March 30, 2017
Supported by NSFC (Grant No. 11201065)

had no common upper bound in the cl-degrees. In [7], Fan and Yu improved this result that for any non-computable Δ_2^0 real α there is a left-c.e. real β such that both of them have no common upper bound in left-c.e. reals. In [6], Downey and Hirschfeldt proved the following result: there is a real (not left-c.e.) which is not cl-reducible to any random real (indeed to any complex real). (In [9], a set A is called *complex* if there is an order (nondecreasing, unbounded, and computable) function h such that $K(A \upharpoonright\upharpoonright x) > h(x)$ for all x.) In [1], Barmpalias and Lewis localized this result that there is a left-c.e. real which is not cl-reducible to any Martin–Löf random left-c.e. real.

The interplay between Turing and stronger reducibilities, such as wtt, cl-reducibility, is a meaningful topic. The following results express that computable Lipschitz reducibility helps understand array-non-computability and the lowness notion. Recall that a degree $\mathbf{d}$ is array non-computable if for any total function $f \leq_{\text{wtt}} \emptyset'$, there is a total function $g \leq_T \mathbf{d}$ not dominated by f; a degree $\mathbf{d}$ is non-low$_2$ if for any total function $f \leq_T \emptyset'$ there is a total function $g \leq_T \mathbf{d}$ not dominated by f; a degree $\mathbf{d}$ is called generalised low$_2$ if $\mathbf{d}'' \leq (\mathbf{d} \vee \mathbf{0}')'$.

Proposition 1.1 ([4]) *For a c.e. degree* $\mathbf{d}$*, the following are equivalent*:

(1) *There are left-c.e. reals* $\alpha, \beta \in \mathbf{d}$ *which have no common upper bound in the cl-degrees of left-c.e. reals.*

(2) *There is a left-c.e. real* $\beta \in \mathbf{d}$ *which is not cl-reducible to any random left-c.e. real.*

(3) *There is a set* $A \in \mathbf{d}$ *which is not cl-reducible to any random left-c.e. real.*

(4) $\mathbf{d}$ *is array non-computable.*

Proposition 1.2 ([4]) *If* $\mathbf{d}$ *is not generalised low*$_2$*, there is some* $A \leq_T \mathbf{d}$ *which is not* ibT-*reducible to any complex real.*

In [8], a uniform version of non-low$_2$-ness was introduced: a Turing degree $\mathbf{d}$ is uniformly non-low$_2$ if for any total function $f \leq_T \emptyset'$, there is a uniform way to define a total function $g \leq_T \mathbf{d}$ such that g is not dominated by f. As a non-trivial subclass of non-low$_2$ degrees, Fan proved that:

Proposition 1.3 ([8]) *If a c.e. degree* $\mathbf{d}$ *is uniformly non-low*$_2$*, then for any non-computable* Δ_2^0 *real* α*, there is a left-c.e. real* $\beta \in \mathbf{d}$ *such that* α *and* β *have no common upper bound in left-c.e. reals under cl-reduciblity.*

Applying the proof method of Proposition 1.3, a c.e. Turing degree $\mathbf{d}$ is non-low$_2$ if and only if for any Δ_2^0 real α, there is a real $\beta \in \mathbf{d}$ such that $\beta \not\leq_{\text{cl}} \alpha$ (see [8]).

In this paper, we continue to study the relation between non-low$_2$-ness and cl-reducibility. We firstly construct two left-c.e. reals β_0 and β_1 which have a property as follows.

Theorem 1.4 *There are two left-c.e. reals* β_0 *and* β_1 *such that, for any left-c.e. real* α*, if* $\beta_0 \leq_{\text{wtt}} \alpha$*, then* $\beta_1 \not\leq_{\text{cl}} \alpha$*.*

If a c.e. degree $\mathbf{d}$ is non-low$_2$, $\mathbf{d}$ can contain β_0 and β_1 in above theorem.

Theorem 1.5 *If a c.e. Turing degree* $\mathbf{d}$ *is non-low*$_2$*, then there are two left-c.e. reals* β_0 *and* β_1 *in* $\mathbf{d}$ *such that, for any left-c.e. real* α*, if* $\beta_0 \leq_{\text{wtt}} \alpha$*, then* $\beta_1 \not\leq_{\text{cl}} \alpha$*.*

Now we can localize Proposition 1.2 in the c.e. Turing degrees.

Corollary 1.6 *If a c.e. Turing degree* $\mathbf{d}$ *is non-low*$_2$*, then there is a left-c.e. real in* $\mathbf{d}$*, which* *is not cl-reducible to any complex* (*wtt-complete*) *left-c.e. real.*

Proof In [9], for any set, it is complex $\Leftrightarrow$ it wtt-computes a DNR function; for any c.e. set, it is wtt-complete $\Leftrightarrow$ it is complex. In [4], for any left-c.e. real γ, there is a c.e. set A such that $A \leq_{\rm cl} \gamma$ and $\gamma \leq_{\rm tt} A$. Then $\gamma =_{\rm wtt} A$. Therefore, γ is wtt-complete $\Leftrightarrow$ A is wtt-complete $\Leftrightarrow$ A wtt-computes a DNR function $\Leftrightarrow$ γ wtt-computes a DNR function $\Leftrightarrow$ γ is complex. Let $\beta_1 \in \mathbf{d}$ be the left-c.e. real in Theorem 1.5, this β_1 is not cl-reducible to any wtt-complete left-c.e. real and so it is not cl-reducible to any complex left-c.e. real. $\square$

But we do not know if the converse of Corollary 1.6 holds, thereby giving a characterization of non-low$_2$-ness in the property of cl-reducibility. We organize the paper as follows: in Section 2, we prove Theorem 1.4; in Section 3, we prove Theorem 1.5 by modifying the proof of Proposition 1.3. Here some notations should be mentioned.

Notation 1.7 For a real α, the digits on the right of the decimal point in its binary expansion are numbered as $1, 2, 3, \ldots$ from left to right; the digits on the left of the decimal point are numbered as $0, -1, -2, \ldots$ from right to left. Symbol $\alpha \restriction_I$ is for those α-digits in interval I; symbol $\alpha(a)$ is for digit a of α; symbol b^k is for a string with k consecutive numbers b; symbol 1^i0^ω is simplified as 1^i.

Notice that if $\beta \leq_{\rm cl} \alpha$ with constant c, then $\beta \leq_{\rm ibT} \alpha \restriction_{\geq c}$. Although we recount our results in cl-reducibility, all "cl" can be substituted by "ibT". To make the technical details of the proofs slightly simpler, we work with ibT-reducibility during the whole construction.

2 One Property of Left-c.e. Reals Under cl-reducibility

To prove Theorem 1.4, it suffices to construct two left-c.e. reals β_0 and β_1, meeting, for all Turing functionals Φ, all ibT-Turing functionals Γ, all left-c.e. reals α, and all partial computable functions f, (w.o.l.g, f is strictly increasing).

$$R_e : \beta_0 = \Phi_{e_2}^{\alpha_{e_1}} \text{ with a computable bound } f_{e_0} \text{ on its use} \Rightarrow \beta_1 \neq \Gamma_{e_3}^{\alpha_{e_1}},$$

where $e = \langle e_0, e_1, e_2, e_3 \rangle$.

2.1 Proof Idea of Theorem 1.4

Inductively, we assume the sequences of intervals $\{I_e\}_{e\in\omega}$ by letting for $e \geq 0$, $x_0 = 1, I_e = [x_e, x'_e) \subset \mathbb{N}$, $x_{e+1} > x'_e$.

Now we assign $\beta_1 \restriction_{I_e}$ and $\beta_0 \restriction_{I_e}$ to meet R_e. From now on, the current value of candidates I_e, x_e etc. at stage s is denoted by putting one more index s. Let

$$l(e,s) = \max\{y : (\forall y < x)[\beta_{0,s}(y) = \Phi_{e_2,s}^{\alpha_{e_1,s}}(y) \ \& \ \phi_{e_2,s}^{\alpha_{e_1,s}}(y) \leq f_{e_0}(y)\downarrow \ \& \ \beta_{1,s}(y) = \Gamma_{e_3,s}^{\alpha_{e_1,s}}(y)]\}.$$

For $s > e$, we say R_e *requires attention* at stage $s+1$ if $l(e,s) \geq x'_{e,s}$. Then we make a change of $\beta_{0,s+1} \restriction_{I_{e,s}}$ or $\beta_{1,s+1} \restriction_{I_{e,s}}$. If R_e requires attention at stage $t > s+1$ again, then either

(1) a change on $\beta_{0,s+1}(x)$ $(x \in I_{e,s})$ will cause $(\alpha_{e_1,s} \neq \alpha_{e_1,t}) \restriction\restriction f_{e_0}(x)$; or

(2) a change on $\beta_{1,s+1}(x)(x \in I_{e,s})$ will cause $(\alpha_{e_1,s} \neq \alpha_{e_1,t}) \restriction\restriction x$.

The idea to meet R_e is: we set I_e long enough and change one appropriate β_i-digit $(i = 0, 1)$ when R_e requires attention. If R_e is not met when $\beta_0 \restriction_{I_e} = \beta_1 \restriction_{I_e} = 1^{x'_e - x_e}$, then the left-c.e. real α_{e_1} will be larger than 1, which is a contradiction.

2.2 The Modules

Now we make preparation for meeting R_e, which shows how to define I_e and arrange β_i-change when R_e requires attention.

Given a computable function f, we describe an ibT-wtt game amongst left-c.e. reals β_0, β_1 and α in stages as follows:

• if β_0 changes on digit x, then α will change on some digit not larger than $f(x)$;

• if β_1 changes on digit x, then α will change on some digit not larger than x.

Following this game, real α wtt-computes β_0 and ibT-computes β_1 simultaneously. In this game, we say α follows the *least effort strategy* to ibT-compute β_1, if α increases by the least amount 2^{-x} when β_1 changes on digit x.

Now we introduce the following lemmas without proof, which is quiet similar to the lemmas in [11].

Lemma 2.1 *In an* ibT-*wtt-game amongst* β_0, β_1 *and* α*,* α *has to follow instructions of the type "change a digit at a position* $\leq f(x)$*" if* β_0 *changes on* x *or "change a digit at a position* $\leq x$*" if* β_1 *changes on* x*. The least effort strategy is the best strategy for* α*. In other words, if a different strategy produces* α'*, then at each stage* s *of the game* $\alpha_s \leq \alpha'_s$*.*

Lemma 2.2 *In the above* ibT-*wtt-game amongst* β_0, β_1 *and* α*, although* $\alpha = 0$*, some* α' *plays the same game while starting with* $\alpha'_0 = \sigma$ *for a finite binary expansion* σ*. If the strategies of* α*,* α' *are the same (i.e. the least effort strategy described above) and the sequence of instructions only ever demand change at digits* $> |\sigma|$*, then at every stage* s*,* $\alpha'_s = \alpha_s + \sigma$*.*

Suppose that α computes β_0 with a computable bound f on its use; let $\mathbb{A} = \{m, m + 2^{-1} : m \in \mathbb{N}^+\}$ and $n \in \mathbb{A}$; and $\beta_0 \upharpoonright_{\geq a} = \beta_1 \upharpoonright_{\geq a} = \alpha \upharpoonright_{\geq a} = 0^\omega$ for $a \in \mathbb{N}^+$. We introduce a module $M(f, n, a)$ to define β_i $(i = 0, 1)$ by induction on n, and during it,

• only digits $\geq a$ of β_i change;

• if some β_i-change occurs, then α wtt-computes β_0 with a computable use f or follows the least strategy to ibT-compute β_1 instantly;

• if one β_i-change occurs, then until α codes this change no other β_i-changes occur;

• the left-c.e. real α satisfies $\alpha \geq n \cdot 2^{-a+1}$ when it ends.

Module 2.3 Given a computable function f, $\beta_0 \upharpoonright_{\geq a} = \beta_1 \upharpoonright_{\geq a} = \alpha \upharpoonright_{\geq a} = 0^\omega$. The module $M(f, 1, a)$ is as follows:

(1) Let $b = a + 1$;

(2) Let $\beta_1(b) = 1$;

(3) (a) If $b < f(a)$, then increase b by amount 1 and go back to (2);

(b) If $b = f(a)$, then go to (4);

(4) Let $\beta_0(a) = 1$;

(5) Let $\beta_1(a) = 1$ and clear all digits $> a$ to 0; end this module.

Lemma 2.4 *Following* $M(f, 1, a)$*,* $\alpha \geq 2^{-a+1}$ *and* $\alpha \upharpoonright_{\geq a} = 0^\omega$*.*

Proof If let $\beta_1(b) = 1$, then to code it, $\alpha(b) = 1$. By induction on b, if $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^{f(a)-a}$, then $\alpha \upharpoonright_{>a} = 1^{f(a)-a}$. If let $\beta_0(a) = 1$, then α changes at a digit $\leq f(a)$, which causes $\alpha \geq 2^{-a}$. If let $\beta_1(a) = 1$, then $\alpha \geq 2^{-a+1}$ and $\alpha \upharpoonright_{\geq a} = 0^\omega$. □

Module 2.5 Given a computable function f, $\beta_0 \upharpoonright_{\geq a} = \beta_1 \upharpoonright_{\geq a} = \alpha \upharpoonright_{\geq a} = 0^\omega$. Let $n \geq 1$ and digit $-k$ be the leftmost non-zero digit in n's binary representation. Suppose that $M(f, n, x)$ is well defined for each $x \in \mathbb{N}^+$, the module $M(f, n + 2^{-1}, a)$ is as follows:

(1) Let $i = 1$;

(2) Perform the module $M(f, n, a + i)$;

(3) When $M(f, n, a + i)$ ends,

(a) If $i < f(2^{k+1} \cdot a) + k$, then increase i by amount 1 and go back to (2);

(b) If $i = f(2^{k+1} \cdot a) + k$, then go to (4);

(4) Let $\beta_0(a) = 1$ and clear all digits $> a$ to 0;

(5) Let $\beta_1(a) = 1$ and clear all digits $> a$ to 0; end this module.

Lemma 2.6 *Following* $M(f, n + 2^{-1}, a)$, $\alpha \geq (n + 2^{-1}) \times 2^{-a+1}$ *and* $\alpha \upharpoonright_{\geq a} = 0^\omega$.

Proof We prove by induction on n.

Case $n = 1$ Firstly, we claim that: if $M(f, 1, a + i)$ ends, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^i$, $\alpha \geq 2^{-a} + \cdots + 2^{-a-i+1}$ and $\alpha \upharpoonright_{\geq a+i+1} = 0^\omega$.

Fix i, assume that this claim holds for $\leq i$. If $M(f, 1, a+i)$ ends, then $\beta_0 \upharpoonright_{>a+i} = \beta_1 \upharpoonright_{>a+i} = 0^\omega$ provides room to perform $M(f, 1, a + i + 1)$. By Lemmas 2.2 and 2.4, if $M(f, 1, a + i + 1)$ ends, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^{i+1}$, $\alpha \geq 2^{-a} + \cdots + 2^{-a-i}$ and $\alpha \upharpoonright_{\geq a+i+2} = 0^\omega$.

For $n = 1$, $k = 0$. Following the claim, if $i = f(2a)$, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^{f(2a)}$ and $\alpha \geq 2^{-a} + \cdots + 2^{-a-f(2a)+1}$. If let $\beta_0(a) = 1$, then to code it, α changes at a digit $\leq f(a)$. For $a + f(2a) - 1 \geq f(a)$, it causes

$$\alpha \geq 2^{-a} + \cdots + 2^{-f(a)} + 2^{-f(a)} \geq 2^{-a+1}.$$

If let $\beta_1(a) = 1$, then to code it, $\alpha \geq 2^{-a+1} + 2^{-a} = (1 + 2^{-1}) \cdot 2^{-a+1}$.

Case $n > 1$ Assume that Lemma 2.6 holds for all $m < n, a \in \mathbb{N}^+$. Firstly, we claim that: if $M(f, n, a+i)$ ends, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^i$, $\alpha \geq n \cdot (2^{-a} + \cdots + 2^{-a-i+1})$ and $\alpha \upharpoonright_{\geq a+i+1} = 0^\omega$.

Fix i, assume that this claim holds for $\leq i$. If $M(f, n, a+i)$ ends, then $\beta_0 \upharpoonright_{>a+i} = \beta_1 \upharpoonright_{>a+i} = 0^\omega$ provides room to perform $M(f, n, a + i + 1)$. By Lemma 2.2, if $M(f, n, a + i + 1)$ ends, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^{i+1}$, $\alpha \geq n \cdot (2^{-a} + \cdots + 2^{-a-i+1} + 2^{-a-i})$ and $\alpha \upharpoonright_{\geq a+i+2} = 0^\omega$.

Following the claim, if $i = f(2^{k+1} \cdot a) + k$, then $\beta_0 \upharpoonright_{>a} = \beta_1 \upharpoonright_{>a} = 1^{f(2^{k+1} \cdot a)+k}$ and

$$\begin{aligned}
\alpha &\geq n \cdot 2^{-a} + \cdots + n \cdot 2^{-a-f(2^{k+1} \cdot a)-k+1} \\
&= 2^{-a+1} \cdot n \cdot (1 - 2^{-f(2^{k+1} \cdot a)-k}) \\
&= (n - 2^{-1}) \cdot 2^{-a+1} + (2^{-a-1} + 2^{-a-2} + \cdots + 2^{-a-f(2^{k+1} \cdot a)+2}) \\
&\quad + (2^{-a-f(2^{k+1} \cdot a)+2} - n \cdot 2^{-a-f(2^{k+1} \cdot a)-k+1}).
\end{aligned}$$

Note that $2^{-a-f(2^{k+1} \cdot a)+2} - n \cdot 2^{-a-f(2^{k+1} \cdot a)-k+1} \geq 0$, since $n \leq 2^{k+1}$.

If let $\beta_0(a) = 1$, then to code it, α changes at a digit $\leq f(a)$. For $a + f(2^{k+1} \cdot a) - 2 \geq f(a)$, it causes

$$\alpha \geq (n - 2^{-1}) \cdot 2^{-a+1} + (2^{-a-1} + 2^{-a-2} + \cdots + 2^{-f(a)}) + 2^{-f(a)} = n \cdot 2^{-a+1}.$$

If let $\beta_1(a) = 1$, then to code it, $\alpha \geq (n + 2^{-1}) \cdot 2^{-a+1}$ and $\alpha \upharpoonright_{\geq a} = 0$. □

Remark 2.7 Let $I_{f,n,a}$ be the least interval which includes all digits of β_0 and β_1 mentioned during $M(f,n,a)$. By induction on n, the first changing digit in $M_{f,n,a}$ is $\beta_1(a+2n-1)$.

2.3 The Construction of β_0 and β_1 in Theorem 1.4

Now we recall R_e-requirement. If $\beta_0 = \Phi_{e_0}^{\alpha_{e_1}}$ with a computable bound f_{e_0} on its use, then we perform $M(f_{e_0}, 2^{x_e}, x_e)$ to make one change of β_0 or β_1 each time R_e requires attention. By Lemmas 2.1, 2.2 and 2.6, R_e is met. Otherwise $\alpha_{e_1} \geq 2^{x_e} \cdot 2^{-(x_e-1)} = 2$, which is a contradiction. However we can not tell whether f_{e_0} is total or not in advance. The following notation helps us to judge whether take f_{e_0} as a (total) computable function.

Notation 2.8 We define a set $D_{f,n,a}$ by induction as follows:

(1) $D_{f,1,a} = \{a : f(a)\downarrow\}$ (we say $D_{f,1,a}$ is well defined at stage s if $f_s(a)\downarrow$).

(2) Suppose that $D_{f,n,x}$ is defined for $x \in \mathbb{N}^+$ and digit $-k$ is the leftmost non-zero digit in n's binary representation. Let

$$D_{f,n+2^{-1},a} = \bigcup_{1\leq i\leq f(2^{k+1}\cdot a)+k} D_{f,n,a+i}.$$

(We say $D_{f,n+2^{-1},a}$ is well defined at stage s if for all $i \in [1, f(2^{k+1}\cdot a)+k]$, $D_{f,n,a+i}$ is well defined at stage s.)

Given f_{e_0}, if $D_{f_{e_0},2^{x_e},x_e}$ is well defined, then we take f_{e_0} as a computable bound (Although it may be non-computable.); otherwise, f_{e_0} is non-computable. Therefore, before performing $M(f_{e_0}, 2^{x_e}, x_e)$ to meet R_e, we ask for whether $D_{f_{e_0},2^{x_e},x_e}$ is well defined or not. Furthermore, we show how to set the length of $I_{e,s} = [x_{e,s}, x'_{e,s})$.

Notation 2.9 At stage s, if $D_{f_{e_0},2^{x_{e,s}},x_{e,s}}$ is well defined, then we assign $I_{e,s}$ such that $|I_{e,s}| = |I_{f_{e_0},2^{x_{e,s}},x_{e,s}}|$; otherwise, we assign $I_{e,s}$ such that $|I_{e,s}| = 2^{x_{e,s}+2}$ (By Remark 2.7, this assures that if $D_{f_{e_0},2^{x_{e,s}},x_{e,s}}$ is well defined and R_e requires attention at stage $s+1$, then we can perform the first action of $M(f_{e_0}, 2^{x_{e,s}}, x_{e,s})$ and reset $I_{e,s+1}$.).

Now we effectively order R_e-requirements in an increasing order of e, and give the construction in stages as follows.

Construction 2.10 Stage $s=0$. Let $\beta_{0,0} = \beta_{1,0} = 0, x_{0,0} = 1$.

Stage $s+1$. Choose the least $e \leq s$ so that R_e requires attention and $D_{f_{e_0},2^{x_{e,s}},x_{e,s}}$ is well defined. Following $M(f_{e_0}, 2^{x_{e,s}}, x_{e,s})$, we make a corresponding change of $\beta_{i,s}$, $i=0,1$, keep $x_{e,s+1} = x_{e,s}$, and set $I_{e,s+1}$ such that $|I_{e,s+1}| = |I_{f_{e_0},2^{x_{e,s}},x_{e,s}}|$. Meanwhile, if $e' < e$, keep $x_{e',s+1} = x_{e',s}$ and $I_{e',s+1} = I_{e',s}$ for stronger requirement $R_{e'}$; and if $e' > e$, then initialize all weaker requirement $R_{e'}$, i.e., reassign $x_{e',s+1}$ to be larger than any number mentioned before, and set $I_{e',s+1}$ such that $|I_{e',s+1}| = 2^{x_{e',s+1}+2}$, consecutively in an increasing order of e'. Otherwise, if there is no such e, then keep all candidates unchanged.

Lemma 2.11 *Fix e, requirement R_e receives attention at most finitely often and is eventually satisfied.*

Proof Fix e and assume by induction that Lemma 2.11 holds for all $i < e$. Choose t minimal so that no R_i, $i < e$, receives attention after stage t. Hence, requirement R_e is not initialized after stage t and $x_e = x_{e,t}$.

Choose the least stage $s > t$ such that $D_{f_{e_0},2^{x_{e,s}},x_{e,s}}$ is well defined. If no such s exists, then f_{e_0} is not total and R_e is met. Otherwise, if R_e is not met, then by Lemmas 2.1, 2.2 and 2.6, $\alpha_{e_1} \geq 2^{x_e} \cdot 2^{-(x_e-1)} > 1$ after $M(f_{e_0}, 2^{x_e}, x_e)$ ends. It is a contradiction. □

Remark 2.12 The interval I_e for R_e is fixed forever after some stage. Since I_e is initialized finitely often and after that it changes only at the first stage when $D_{f_{e_0},2^{x_e},x_e}$ is well defined and R_e requires attention. Hence $\{I_e\}_{e\in\omega}$ is T-reducible to $\emptyset'$.

3 Non-low$_2$-ness: cl-reducibility

To prove Theorem 1.5, we can construct two left-c.e. reals β_0, β_1 to meet the following requirements: for $i = 0, 1$, $P_i : \beta_i \leq_T D$ and $T_i : D \leq_T \beta_i$; and for $e \in \omega$,

$$Q:\ \exists^\infty e\Big[\ Q_e = \bigwedge_{0\leq i\leq e} Q_{e,i}\Big],$$

which $Q_{e,i} = R_i$. Effectively order $Q_{e,i}$-requirements (of order type ω) as follows: $Q_{e,i} < Q_{e',i'}$ if $e < e'$ or $e = e', i < i'$.

Inspired by the proof of Theorem 1.4, let $I_{e,i} = [x_{e,i}, x'_{e,i})$, we perform $M(f_{i_0}, 2^{x_{e,i}}, x_{e,i})$ to change $\beta_0 \restriction_{I_{e,i}}$ or $\beta_1 \restriction_{I_{e,i}}$ to meet $Q_{e,i}$. Let $\mathbf{x}_e = x_{e,0}, \mathbf{x}'_e = x'_{e,e}, \mathbf{x}_{e+1} = \mathbf{x}'_e + 2$, we define

$$\mathbb{I}_e = \bigcup_{0\leq i\leq e} I_{e,i} = [\mathbf{x}_e, \mathbf{x}'_e].$$

To meet T_i we define $\beta_i(\mathbf{x}'_e + 1) = 1$ if $e \in D$; otherwise, define $\beta_i(\mathbf{x}'_e + 1) = 0$.

In this way, both Q-requirements and T-requirements are met. Hence $\{\mathbb{I}_e\}_{e\in\omega}$ can be considered as the given one.

Now to meet P_i, we define

$$m(e) = \mu s(\forall i \leq e)[\alpha_{i,s} \restriction\restriction \mathbf{x}'_{e,s} = \alpha_i \restriction\restriction \mathbf{x}'_e].$$

By Remark 2.12, $\{\mathbf{x}'_e\}_{e\in\omega}$ is T-reducible to $\emptyset'$ by the Limit Lemma. Hence, m is T-reducible to $\emptyset'$. Let D be a c.e. set in $\mathbf{d}$, there is a total function Ψ^D so that $\exists^\infty e[\Psi^D(e) > m(e)]$ since $\mathbf{d}$ is non-low$_2$. We define a function g as follows: let $g_e(e) = 0$; for $e < s$, $g_s(e) = s$ if $\Psi^D_s(e)\downarrow\ \&\ D_s \restriction \psi^D_{s-1}(e) \neq D_{s-1} \restriction \psi^D_{s-1}(e)$; otherwise, $g_s(e) = g_{s-1}(e)$. We change $\beta_0 \restriction_{\mathbb{I}_e}$ or $\beta_1 \restriction_{\mathbb{I}_e}$ only if $g_s(e) \neq g_{s-1}(e)$. Notice that $g \leq_T D$ for D is c.e., and it implies $\beta_i \leq_T D$. Then from the given $\{\mathbb{I}_e\}_{e\in\omega}$ and m, we begin our construction by the assumption that g is given.

Remark 3.1 Notice that if $Q_{e,i}$ requires attention at stage s but $g_s(e) = g_{s-1}(e)$, then the change of $\beta_0 \restriction_{\mathbb{I}_{e,i}}$ or $\beta_1 \restriction_{\mathbb{I}_{e,i}}$, following $M(f_{i_0}, 2^{x_{e,i}}, x_{e,i})$, does not occur at stage s. This β_i-change will occur at stage $t > s$, which $g_t(e) \neq g_{t-1}(e)$ and $Q_{e,i}$ requires attention. Therefore, to assure $\beta_i \leq_T D$, performing $M(f_{i_0}, 2^{x_{e,i}}, x_{e,i})$ to meet $Q_{e,i}$ slows down. Moreover, although $I_{e,i}$ is given, we need to assure that $\beta_0 \restriction_{I_{e,i}} = \beta_1 \restriction_{I_{e,i}} = 0^{|I_{e,i}|}$ if we want to perform $M(f_{i_0}, 2^{x_{e,i}}, x_{e,i})$. To assure this happens may ask for $Q_{e'}$ ($e' < e$) to initialize Q_e, but $g(e')$ may not change instantly to allow such initialization. Therefore in the following construction, according to g we will define different candidates $\hat{I}_{e,i}, \hat{\mathbb{I}}, \hat{x}_{e,i}, \hat{x'}_{e,i}, \hat{\mathbf{x}}_e$ in stages etc. to help meet $Q_{e,i}$. The current value of all these candidates at stage s is denoted by putting one more index s. If f_{i_0} is a computable bound, for $s > e$, we say $Q_{e,i}$ *requires attention* at stage $s + 1$ if $l(i, s) \geq \hat{\mathbf{x}}'_{e,s}$.

Construction 3.2 We give the construction in stages.

Stage $s=0$. Let $\beta_{0,0}=\beta_{1,0}=0$, and $\hat{x}_{0,0,0}=1$.

Stage $s+1$. Choose the least $\langle e,i\rangle \leq s$ so that $Q_{e,i}$ requires attention, $D_{f_{i_0},2^{\hat{x}_{e,i,s}},\hat{x}_{e,i,s}}$ is well defined and $g_{s+1}(e)\neq g_s(e)$. Following $M(f_{i_0},2^{\hat{x}_{e,i,s}},\hat{x}_{e,i,s})$, we make a corresponding change of β_0 and β_1, keep $\hat{x}_{e,i,s+1}=\hat{x}_{e,i,s}$, and set $\hat{I}_{e,i,s+1}$ such that $|\hat{I}_{e,i,s+1}|=|\hat{I}_{f_{i_0},2^{\hat{x}_{e,i,s+1}},\hat{x}_{e,i,s+1}}|$. Moreover, if $e'<e$ or $e'=e, i'<i$, then keep $\hat{x}_{e',i',s+1}=\hat{x}_{e',i',s}, \hat{I}_{e',i',s+1}=I_{e',i',s}$ for stronger requirement $Q_{e',i'}$; and if $e<e'$ or $e=e', i<i'$, then initialize weaker requirement $Q_{e',i'}$, i.e., reassign $\hat{x}_{e',i',s+1}$ to be larger than any number mentioned before, keep $\hat{\mathbf{x}}_{e+1,s+1}=\hat{\mathbf{x}}'_{e,s+1}+2$, and set $\hat{I}_{e',i',s+1}$ such that $|\hat{I}_{e',i',s+1}|=2^{\hat{x}_{e',i',s+1}+2}$, consecutively in an increasing order of e',i'. Or else, if there is no such $\langle e,i\rangle$, then keep all candidates unchanged.

Furthermore, for $i=0,1$, we define $\beta_{i,s+1}(\hat{\mathbf{x}}'_{e,s+1}+1)=1$ if $e\in D_{s+1}$; otherwise, let $\beta_{i,s+1}(\hat{\mathbf{x}}'_{e,s+1}+1)=0$.

Lemma 3.3 *For each e and $0\leq i\leq e$, $\hat{x}_{e,i}\leq x_{e,i}$ and $\hat{x}'_{e,i}\leq x'_{e,i}$.*

Proof Fix e,i and assume by induction that Lemma 3.3 holds for all $e'<e$ or $e'=e$, $i'<i$. It is obvious that $\hat{x}_{e,i}\leq x_{e,i}$. By Module 2.5 and Lemma 2.6, if f_i is strictly increasing, then $|I(f_i,2^{\hat{x}_{e,i}},\hat{x}_{e,i})|\leq |I(f_i,2^{x_{e,i}},x_{e,i})|$. Hence $\hat{x}'_{e,i}\leq x'_{e,i}$. □

Lemma 3.4 *Q is met, i.e. $\exists^\infty e[Q_e$ is met$]$.*

Proof Fix $Q_{e,i}$, choose t minimal so that no stronger requirements than $Q_{e,i}$ ask for β_0-change or β_1-change or no $i<e$ is enumerated into D. Hence, requirement $Q_{e,i}$ is not initialized after stage t and $\hat{x}_{e,i}=\hat{x}_{e,i,t}$. Choose the least stage $s>t$ such that $D_{f_{i_0},2^{\hat{x}_{e,i}},\hat{x}_{e,i}}$ is well defined. If no such s exists, then f_{i_0} is not total and $Q_{e,i}$ is met. If such s exists, then $\hat{I}_{e,i}=\hat{I}_{e,i,s}$. If $Q_{e,i}$ requires attention after s, then we follow $M(f_{i_0},2^{\hat{x}_{e,i,s}},\hat{x}_{e,i,s})$ to change $\beta_0\restriction_{\hat{I}_{e,i}}$ or $\beta_1\restriction_{\hat{I}_{e,i}}$. By Lemma 3.3, $\hat{\mathbf{x}}'_e\leq \mathbf{x}_e$. So $\alpha_{i,m(e)}\restriction\restriction \hat{\mathbf{x}}'_{e,m(e)}=\alpha_i\restriction\restriction \hat{\mathbf{x}}'_e$. We observe whether $Q_{e,i}$ requires attention at stage $m(e)>s$. If $Q_{e,i}$ does not require attention at $m(e)$, then $Q_{e,i}$ is met. If $Q_{e,i}$ requires attention at $m(e)$, then we continue to perform $M(f_{i_0},2^{\hat{x}_{e,i}},\hat{x}_{e,i})$ after $m(e)$, since $M(f_{i_0},2^{\hat{x}_{e,i}},\hat{x}_{e,i})$ will not end during the whole construction. Otherwise, if $M(f_{i_0},2^{\hat{x}_{e,i}},\hat{x}_{e,i})$ ends, then $\alpha_{i_1}\geq 2^{\hat{x}_{e,i}}\cdot 2^{-(\hat{x}_{e,i}-1)}>1$ by Lemmas 2.1, 2.2 and 2.6, which is a contradiction. In this way, $Q_{e,i}$ is met at stage $g(e)$ forever if $g(e)>m(e)$. For $\forall e[g(e)\geq \Phi^D(e)]$ and $\exists^\infty e[\Phi^D(e)>m(e)]$, there are infinitely many e such that $g(e)>m(e)$ and Q_e is met. □

Lemma 3.5 $\beta_i\equiv_T D\in \mathbf{d}$.

Proof It is obvious that $D\leq_T \beta_i$. For $x\in \hat{\mathbb{I}}_e$, $\beta_{i,s+1}(x)$ changes implies $g_{s+1}(e)$ does. For D is c.e., $g\leq_T D$. Then $\beta_i\leq_T D$. □

Acknowledgements We thank the referees for their time and comments.

References

[1] Barmpalias, G., Lewis, A.: A left-c.e. real that cannot be *SW*-computed by any Ω number. *Notre Dame J. Formal Logic*, **47**(2), 197–209 (2006)

[2] Barmpalias, G., Lewis, A.: Random reals and Lipschitz continuity. *Math. Structures Computer Sci.*, **16**, 737–749 (2006)

[3] Barmpalias, G., Lewis, A.: Randomness and the linear degrees of computability. *Ann. Pure Appl. Logic*, **145**, 252–257 (2007)

[4] Barmpalias, G., Downey R., Greenberg, N.: Working with strong reducibilities above totally ω-c.e. degrees and array computable degrees. *Trans. Amer. Math. Soc.*, **362**(2), 777–813 (2010)

[5] Downey, R., Hirschfeldt, D., LaForte, G.: Randomness and reducibility. Mathematical Foundations of Computer Science 2001, Lecture Notes in Computer Science, **2136**, 316–327 (2001); Final version in Journal of Computing and System Sciences, **68**, 96–114 (2004)

[6] Downey, R., Hirschfeldt, D.: Algorithmic Randomness and Complexity, Springer-Verlag Monographs in Computer Science, New York, 2010

[7] Fan, Y., Yu, L.: Maximal pairs of left-c.e. reals in the computably Lipschitz degrees. *Ann. Pure Appl. Logic*, **162**(5), 357–366 (2011)

[8] Fan, Y.: A uniform version of non-low$_2$-ness. *Ann. Pure Appl. Logic*, **168**(3), 738–748 (2017)

[9] Kjos-Hanssen, K., Merkle, W., Stephan, F.: Kolmogorov complexity and the recursion theorem. Twenty-Third Annual Symposium on Theoretical Aspects of Computer Science, Marseille, France, February 23–25, 2006. *Proceedings, Springer Lecture Notes in Computer Science*, **3884**, 149–161 (2006)

[10] Soare, R.: Computability theory and differential geometry. *Bull. Symbolic Logic*, **10**(4), 457–486 (2004)

[11] Yu, L., Ding, D.: There is no SW-complete left-c.e. real. *J. Symbolic Logic*, **69**(4), 1163–1170 (2004)