QUANTIFIER VARIATIONS IN SOLOVAY REDUCIBILITY

MASAHIRO KUMABE, KENSHI MIYABE, AND TOSHIO SUZUKI

ABSTRACT. Solovay reducibility, a fundamental concept in algorithmic randomness, is used to study the relative randomness of real numbers. This paper examines how changing quantifiers in the definition of Solovay reducibility affects its application to left-c.e. reals and computably approximable (c.a.) reals. For left-c.e. reals, using a universal quantifier for the first sequence yields an equivalent notion of reducibility. However, for c.a. reals, this equivalence holds only when considering computable subsequences. Our study focuses on this difference observed in c.a. reals as its central result. Additionally, we propose a more robust definition of Solovay reducibility for c.a. reals.

CONTENTS

Date: 2025-04-12 18:15:17 JST.

2020 *Mathematics Subject Classification.* 03D32,68Q30.

Key words and phrases. Solovay reducibility, priority argument.

1. Introduction

Solovay reducibility, a key concept in algorithmic randomness, allows us to compare the randomness of real numbers by looking at how quickly they can be approximated. It has become an essential tool for understanding how real numbers are structured in computability theory. This paper explores variations in characterizations of this important concept, focusing on how different quantifier choices affect its application to various types of real numbers.

Originally introduced by Solovay [6], the concept involves a relation between two reals, α and β, using a function from $\mathbb{Q}$ to $\mathbb{Q}$ that computably transforms an approximation of β to that of α (Definition 2.1). Later, Downey et al. [1, Lemma 2.3] characterized Solovay reducibility for left-c.e. reals using increasing computable approximations $(a_n)_n$ and $(b_n)_n$ of α and β respectively (Proposition 2.2).

For left-c.e. reals, using universal ($\forall$) or existential ($\exists$) quantifiers for $(a_n)_n$ and $(b_n)_n$ often leads to equivalent concepts (see Section 2). However, the original definition of Solovay reducibility works well for left-c.e. reals but not for other reals [2, Proposition 9.6.1].

Zheng and Rettinger [7] introduced a new Solovay reducibility for computably approximable (c.a.) reals, which coincides with the original concept for left-c.e. reals and behaves better for c.a. reals [5]. The new notion is denoted by $\leq_S^{2a}$ in the original paper [7], but we just call it Solovay reducibility and denote it by $\leq_S$. Their definition uses $\exists$ for both $(a_n)_n$ and $(b_n)_n$ (Definition 3.1).

Some researchers believe this robustness also applies to c.a. reals (Remark 3.2). However, this paper aims to show that this is not true. We demonstrate that altering an existential quantifier ($\exists$) to a universal quantifier ($\forall$) in the definition of Solovay reducibility for c.a. reals leads to a different notion (Theorems 4.1 and 5.1), which are main results of this paper.

The structure of this paper is as follows: In Section 2, we present observations on Solovay reducibility for left-c.e. reals. We consider some quantifier variations in Solovay reducibility and prove that some of them are equivalent and others are not. In Section 3, we extend our analysis to c.a. reals, examining similar quantifier variations in Solovay reducibility for this broader class of reals. We investigate which variations remain equivalent and which lead to distinct notions in this context. In Section 4 and 5, we demonstrate the non-equivalence that arises when altering quantifiers, supported by specific counterexamples. In Section 6, we introduce a more robust definition of Solovay reducibility for c.a. reals, addressing the issues identified in previous sections.

2. Variants of Solovay reducibility for left-c.e. reals

In this section, we aim to demonstrate the equivalence or non-equivalence of quantifier variations, using either "exists" (∃) or "forall" (∀), in the characterization of Solovay reducibility for left-c.e. reals based on computable approximations by sequences.

2.1. **Definition and characterization.** Let **EC**, **LC** and **CA** be the set of all computable, left-c.e., and computably approximable reals, respectively. For $\alpha \in \mathbf{CA}$, let $\mathrm{CS}(\alpha)$ be the set of all computable sequences of rationals converging to α. For $\alpha \in \mathbf{LC}$, let $\mathrm{ICS}(\alpha)$ be the set of all increasing computable sequences of rationals converging to α. For $\alpha \in \mathbf{LC}$, let $\mathrm{ICS}_0(\alpha)$ be the set of all nondecreasing computable sequence of rationals converging to α. Let S be the set of all nondecreasing unbounded computable functions from ω to ω. For ease of reference, we summarize these definitions below in Table 1:

Set	Description
EC	Computable reals
LC	Left-c.e. reals
CA	Computably approximable reals
$\mathrm{CS}(\alpha)$	Computable sequences $\to \alpha$
$\mathrm{ICS}(\alpha)$	Increasing computable sequences $\to \alpha$
$\mathrm{ICS}_0(\alpha)$	Nondecreasing computable sequences $\to \alpha$
S	Nondecreasing unbounded computable sequences

Table 1. Definition of sets

The original definition of Solovay reducibility uses a partial computable function that transforms an approximation of one real to an approximation of the other real.

Definition 2.1 (Solovay [6]). Let $\alpha, \beta \in \mathbf{LC}$. We say that α is *Solovay reducible* to β, denoted by $\alpha \leq_S \beta$, if there are a partial computable function $f :\subseteq \mathbb{Q} \to \mathbb{Q}$ and $c \in \omega$ such that, if $q \in \mathbb{Q}$ and $q < \beta$, then $f(q) \downarrow < \alpha$ and $\alpha - f(q) < c(\beta - q)$.

Solovay reducibility for left-c.e. reals can be characterized by computable sequences approaching α and β. We will primarily use the characterization below in this section.

Proposition 2.2 (Downey et al. [1, Lemma 2.3]). *Let $\alpha, \beta \in \mathbf{LC}$, $(a_s)_s \in \mathrm{ICS}(\alpha)$, and $(b_s)_s \in \mathrm{ICS}(\beta)$. Then, $\alpha \leq_S \beta$ if and only if there are a function $g \in S$ and a constant $c \in \omega$ such that $\alpha - a_{g(s)} < c(\beta - b_s)$ for all $s \in \omega$.*

2.2. **Variations.** Solovay reducibility for left-c.e. reals is known to be somewhat robust even under changes in quantification. In what follows, we will enumerate (almost) all possible scenarios.

Let $\alpha, \beta \in \mathbf{LC}$, $(a_s)_s \in \mathrm{ICS}(\alpha)$, and $(b_s)_s \in \mathrm{ICS}(\beta)$. Let P_0^L, P_a^L, P_b^L be the relations of them defined by the following:

$$P_0^L :(\exists q \in \omega)(\forall s \in \omega)[\alpha - a_s < q(\beta - b_s)], \tag{1}$$

$$P_a^L :(\exists g \in S)(\exists q \in \omega)(\forall s \in \omega)[\alpha - a_{g(s)} < q(\beta - b_s)], \tag{2}$$

$$P_b^L :(\exists g \in S)(\exists q \in \omega)(\forall s \in \omega)[\alpha - a_s < q(\beta - b_{g(s)})]. \tag{3}$$

Then, we consider the following conditions for $\alpha, \beta \in \mathbf{LC}$:

(L-I) $(\exists(a_s))(\exists(b_s))P_0^L$,
(L-II) $(\forall(b_s))(\exists(a_s))P_0^L$,
(L-III) $(\forall(a_s))(\exists(b_s))P_0^L$,
(L-IV) $(\exists(b_s))(\forall(a_s))P_0^L$,
(L-V) $(\exists(a_s))(\forall(b_s))P_0^L$,
(L-VI) $(\forall(a_s))(\forall(b_s))P_0^L$.

We also consider the following subsequence versions:

(L-VI-S1) $(\forall(a_s))(\forall(b_s))P_a^L$,
(L-VI-S2) $(\forall(a_s))(\forall(b_s))P_b^L$.

In this context, we assume $(a_s)_s \in \mathrm{ICS}(\alpha)$ and $(b_s)_s \in \mathrm{ICS}(\beta)$ when considering the quantifiers.

The goal of this section is to show that $\alpha \leq_S \beta$ and each condition (L-I), (L-II), (L-III), (L-VI-S1), and (L-VI-S2) are mutually equivalent, and each condition (L-IV), (L-V), and (L-VI) is not equivalent to $\alpha \leq_S \beta$. The order of the proofs is illustrated in Figure 1.

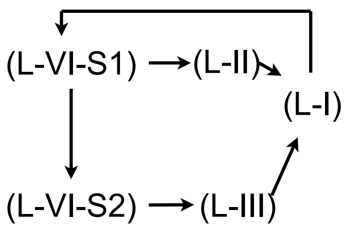

FIGURE 1. The order of the proofs

Proposition 2.3. *Let $\alpha, \beta \in \mathbf{LC}$. Then, $\alpha \leq_S \beta$ and each condition (L-I), (L-II), and (L-VI-S1) are mutually equivalent.*

Proof. Proposition 2.2 states that $\alpha \leq_S \beta$ is equivalent to condition (L-VI-S1).

Assume condition (L-VI-S1). Since $(a_s)_s$ is increasing, we can further impose that the function $g \in S$ is strictly increasing. Thus, condition (L-II) holds.

The fact that condition (L-II) implies condition (L-I) is immediate from the definition.

Assume condition (L-I) and let $(a_s)_s$ and $(b_s)_s$ be its witness. By considering Proposition 2.2 for this pair, the condition is met by letting g be the identity function and c as q, which concludes $\alpha \leq_S \beta$. □

Proposition 2.4. *Let $\alpha, \beta \in$ **LC**. Then, each condition (L-I), (L-III), (L-VI-S2) are mutually equivalent.*

Proof. (L-VI-S1) $\Longrightarrow$ (L-VI-S2).

Let $(a_s)_s \in \mathrm{ICS}(\alpha)$ and $(b_s)_s \in \mathrm{ICS}(\beta)$. By (L-VI-S1), there exist a function $g \in S$ and $q \in \omega$ with which P_a^L holds. We further assume that $g(0) = 0$ and g is increasing by redefining q as a larger integer if necessary. (This trick is necessary to define $h(0)$ below.) We define a function $h : \omega \to \omega$ by

$$h(t) = \max\{s \ : \ g(s) \leq t\}.$$

The function $h(t)$ is similar to an inverse function in that it finds the largest s for which $g(s)$ does not exceed t. Then, h is nondecreasing, unbounded and computable, thus $h \in S$. Furthermore, by letting $s = h(t)$, we have

$$\alpha - a_t \leq \alpha - a_{g(s)} < q(\beta - b_s) = q(\beta - b_{h(t)}),$$

which implies that P_b^L holds. Thus, the claim is proven.

(L-VI-S2) $\Longrightarrow$ (L-III).

Assume $(a_n)_n \in \mathrm{ICS}(\alpha)$ and $(b_s)_s \in \mathrm{ICS}(\beta)$. By condition (L-VI-S2), there exists a function $g \in S$ with which P_b^L holds.

If g is increasing, then let $d_s = b_{g(s)}$ and we are done. For the case that g is not increasing, the inputs n that yield the same value when g is applied need to be dispersed so that $(d_s)_s$ is increasing as in Figure 2.

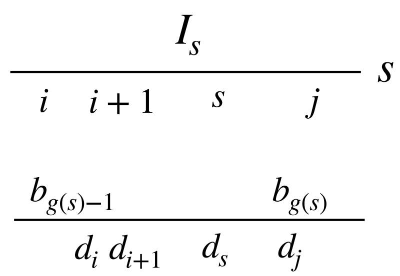

Figure 2. Definition of $(d_s)_s$

For each $s \in \omega$, let $I_s = \{t \ : \ g(s) = g(t)\}$. Note that $g \in S$. Since g is nondecreasing and unbounded, I_s is a finite set for each s. Furthermore, since g is computable, so is I_s. Let $m = |I_s|$ be the number of the elements in I_s and suppose that s is the k-th smallest element in I_s. We define d_s as follows. If $g(s) = 0$, then

$$d_s = b_0 - 1 + \frac{k}{m}.$$

If $g(s) > 0$, then

$$d_s = b_{g(s)-1} + \frac{k}{m}(b_{g(s)} - b_{g(s)-1}).$$

The sequence $(d_s)_s$ is increasing and computable, thus $(d_s)_s \in \mathrm{ICS}(\beta)$. Since $d_s \leq b_{g(s)}$, we have

$$\alpha - a_s < q(\beta - b_{g(s)})) \leq q(\beta - d_s).$$

Thus, condition (L-III) holds.

(L-III) $\Longrightarrow$ (L-I).

This is immediate from the definition. □

Each condition (L-IV), (L-V), (L-VI) is not equivalent to Solovay reducibility as shown below.

Example 2.5. Let $\alpha, \beta \in \mathbf{EC}$. Then, $\alpha \leq_S \beta$ holds. However, none of the conditions (L-IV), (L-V), or (L-VI) holds for this pair of α and β. As (L-V), for example, for any approximation (a_n), we can construct (b_n) the convergence rate of which is much faster.

Remark 2.6. If $\beta \in \mathbf{LC}$ is Martin-Löf random, then the strongest condition (L-VI) holds for any $\alpha \in \mathbf{LC}$ by a result of Kučera and Slaman [3]. See also Miller [4, Lemma 1.1]. Thus, all of the six conditions, from (L-I) to (L-VI), are equivalent in this case.

3. Variants of Solovay reducibility for c.a. reals

After examining quantifier variants of Solovay reducibility for left-c.e. reals in the previous section, we now focus on computably approximable (c.a.) reals.

3.1. **Definition.** The original definition of Solovay reducibility is well-suited for left-c.e. reals but not for other classes of reals; see Section 1 for a reference. Zheng and Rettinger [7] introduced an alternative definition of Solovay reducibility specifically for c.a. reals. This new definition coincides with the original one for left-c.e. reals (Definition 2.1) and behaves better even outside of left-c.e. reals.

Definition 3.1. Let $\alpha, \beta \in \mathbf{CA}$. We say that α is *Solovay reducible* to β, denoted by $\alpha \leq_S \beta$, if there exist sequences $(a_s)_s \in \mathrm{CS}(\alpha)$, $(b_s)_s \in \mathrm{CS}(\beta)$, and a constant $q \in \omega$ such that

$$|\alpha - a_s| < q(|\beta - b_s| + 2^{-s})$$

for all $s \in \omega$.

3.2. **Variants.** We consider corresponding variants of the conditions in the previous sections for c.a. reals. Let P_0, P_a, P_b be the relations defined by the following:

$$\begin{aligned} P_0 &:(\exists q \in \omega)(\forall s \in \omega)[|\alpha - a_s| < q(|\beta - b_s| + 2^{-s})], \\ P_a &:(\exists q \in \omega)(\forall s \in \omega)[|\alpha - a_{g(s)}| < q(|\beta - b_s| + 2^{-s})], \\ P_b &:(\exists q \in \omega)(\forall s \in \omega)[|\alpha - a_s| < q(|\beta - b_{g(s)}| + 2^{-s})]. \end{aligned}$$

Then, we consider the following conditions for $\alpha, \beta \in \mathbf{CA}$:

(I) $(\exists (a_s))(\exists (b_s))P_0$,
(II) $(\forall (b_s))(\exists (a_s))P_0$,
(III) $(\forall (a_s))(\exists (b_s))P_0$,

We also consider the following subsequence versions:

(II-S) $(\forall (b_s))(\exists (a_s))(\exists g \in S)P_b$,
(III-S) $(\forall (a_s))(\exists (b_s))(\exists g \in S)P_a$,

In this context, we assume $(a_s)_s \in \mathrm{CS}(\alpha)$ and $(b_s)_s \in \mathrm{CS}(\beta)$ when considering the quantifiers.

Notice that condition (I) rephrases the definition of Solovay reducibility for c.a. reals. The goal of this section is to show that conditions (I), (II-S), and (III-S) are mutually equivalent. In later sections, we prove that conditions (II) and (III) are not equivalent to condition (I).

Remark 3.2. Rettinger and Zheng [5, Lemma 3.2] incorrectly claimed that conditions (I) and (II) are equivalent without giving details of the proof. The difference between (I) and (II) is subtle, but it requires careful consideration.

Proposition 3.3. *Let $\alpha, \beta \in \mathbf{CA}$. Then, conditions (I) and (II-S) are equivalent.*

Proof. Notice that condition (II-S) implies (I). In the case of left-c.e. reals (L-I), it was necessary for $(b_s)_s$ to be in $\mathrm{ICS}(\beta)$. For c.a. reals, it suffices that $(b_s)_s$ belongs to $\mathrm{CS}(\beta)$. Since $(b_s)_s$ now does not need to be increasing, the claim concludes easily.

Suppose condition (I) holds via $(a_s)_s \in \mathrm{CS}(\alpha)$, $(b_s)_s \in \mathrm{CS}(\beta)$, and $q \in \omega$. Let $(d_s)_s \in \mathrm{CS}(\beta)$ be given. We construct a function $g \in S$. For each $t \in \omega$, pick up $s \in \omega$ such that

$$s \geq t + 1, \quad |b_s - d_s| < 2^{-t-1},$$

and let $g(t)$ be such s. We can further impose that g is strictly increasing and computable. Then,

$$\begin{aligned} |\alpha - a_{g(t)}| < q(|\beta - b_{g(t)}| + 2^{-g(t)}) &\leq q(|\beta - d_{g(t)}| + |b_{g(t)} - d_{g(t)}| + 2^{-g(t)}) \\ &\leq q(|\beta - d_{g(t)}| + 2^{-t}). \end{aligned}$$

Hence, the pair g and $(a_{g(t)})_t$ serves as a witness for condition (II-S). □

Proposition 3.4. *Let $\alpha, \beta \in \mathbf{CA}$. Then, conditions (I) and (III-S) are equivalent.*

The proof is similar to that of Proposition 3.3.

Proof. Notice that condition (III-S) implies condition (I).

Suppose that condition (I) holds via $(a_s)_s \in \mathrm{CS}(\alpha)$, $(b_s)_s \in \mathrm{CS}(\beta)$, and $q \in \omega$. Suppose that $(c_s)_s \in \mathrm{CS}(\alpha)$ is given. For each $t \in \omega$, pick up $s \in \omega$ such that

$$s \geq t+1, \quad |a_s - c_s| < 2^{-t-1},$$

and let $g(t)$ be such s. We can further impose that g is strictly increasing and computable. Then,

$$\begin{aligned}|\alpha - c_{g(t)}| &\leq |\alpha - a_{g(t)}| + 2^{-t-1} \\ &< q(|\beta - b_{g(t)}| + 2^{-g(t)}) + 2^{-t-1} \\ &\leq q(|\beta - b_{g(t)}| + 2^{-t}).\end{aligned}$$

Thus, $(b_{g(t)})_t$ and g is a witness for condition (III-S). □

4. First counterexample

In this section, we prove that condition (II) is strictly stronger than (I). Recall that Proposition 2.3 asserts the equivalence of conditions (L-I) and (L-II) for left-c.e. reals. We have also shown in Proposition 3.3 that (I) and the subsequence version (II-S) of (II) are equivalent. This is a difference between Solovay reducibility for left-c.e. reals and for c.a. reals.

4.1. Claim.

Theorem 4.1. *There exists a pair of $\alpha, \beta \in \mathbf{CA}$ such that condition (I) holds, but condition (II) does not hold. Furthermore, we can impose $\alpha, \beta \in \mathbf{LC}$.*

Remark 4.2. For left-c.e. reals $\alpha, \beta \in \mathbf{LC}$, Solovay reducibility for left-c.e. reals is equivalent to that for c.a. reals; in other words, conditions (L-I) and (I) are equivalent. We have also seen that conditions (L-I) and (L-II) are equivalent. These facts do not contradict the theorem stated above.

We construct $(a_s)_s$ and $(b_s)_s$ for condition (I) and their limits are α and β, respectively:

$$\exists (a_s)_s \in \mathrm{CS}(\alpha) \exists (b_s)_s \in \mathrm{CS}(\beta) \exists q \in \omega \forall s \in \omega [|\alpha - a_s| < q|\beta - b_s| + 2^{-s}].$$

In fact, we impose $\alpha, \beta \in \mathbf{LC}$ and enforce a stronger condition:

$$\exists (a_s)_s \in \mathrm{ICS}_0(\alpha) \exists (b_s)_s \in \mathrm{ICS}_0(\beta) \forall s \in \omega [0 < \alpha - a_s \leq \beta - b_s]. \tag{4}$$

We enforce the negation of condition (II), that is,

$$\exists (d_s)_s \in \mathrm{CS}(\beta) \forall (c_s)_s \in \mathrm{CS}(\alpha) \forall q \in \omega \exists t \in \omega [|\alpha - c_t| \geq q(|\beta - d_t| + 2^{-t})]. \tag{5}$$

We will also construct $(d_s)_s$.

For condition (I) or the stronger condition (4), the range of possible values for α is quite restricted if β is close to some b_s, but the range of possible values of α is relatively wide if β is far away from each of $(b_s)_s$. We select $(d_s)_s$ so that it remains far from $(b_s)_s$. Suppose some c_t is given. Then we force β to be sufficiently close to d_t. Since the possible value of α is relatively wide, we can force conditions (I) and (5) for this $(c_s)_s$.

We require that $(a_s)_s$ and $(b_s)_s$ are nondecreasing, which implies that α, β are left-c.e. In contrast, we cannot further require that $(d_s)_s$ is nondecreasing in condition (5). For contradiction, suppose that $(d_s)_s$ is nondecreasing. We can further assume that $(a_s)_s$, $(b_s)_s$ are increasing by taking subsequences. Let $g \in S$ be such that $(d_{g(s)})_s$ is increasing. By condition (I), we have

$$\alpha - a_s < q(\beta - (b_s - 2^{-s}))$$

for all s. Thus, condition (L-I) holds via $(a_s)_s$ and $(b_s - 2^{-s})_s$. By Proposition 2.3, condition (L-II) holds. Since $(d_{g(s)})_s$ is increasing, there exist a sequence $(c_s)_s \in \mathrm{ICS}(\alpha)$ and $q \in \omega$ such that

$$\alpha - c_s < q(\beta - d_{g(s)})$$

for all $s \in \omega$. By repeating, we can find $(c'_s)_s \in \mathrm{ICS}(\alpha)$ such that

$$\alpha - c'_s < q(\beta - d_s)$$

for all $s \in \omega$, which implies condition (II), a contradiction.

4.2. **Local strategy.** As a warmup, fix a total computable function $f_e : \omega \to \mathbb{Q}$ to possibly denote a sequence $(c_s)_s$ and fix $q_e \in \omega$. We later diagonalize all such pairs.

The strategy has four states.

- Sleeping state: Do nothing.
- Preparing state ($s \leq t$): Choose $t \in \omega$ for a witness of (5) and define a_s, b_s, d_s for $s \leq t$.
- Waiting state ($t < s < u$): Wait until $f_e(t)$ is defined at stage u.
- Forcing state ($s \geq u$): Select a new forcing region D, and adjust the moving area of a_s, b_s, d_s accordingly.

The initial forcing region is the unit square $D = [x_0, x_1] \times [y_0, y_1] = [0, 1] \times [0, 1]$.

Sleeping state.

Wait until all requirements with higher priorities has become waiting or forcing states. We are currently considering a single strategy, and the strategy is not sleeping at any stage.

Preparing state.

The main task in this state is to choose t and wait until stage t. The number t will be the witness of (5) and we need to put d_t in an appropriate place.

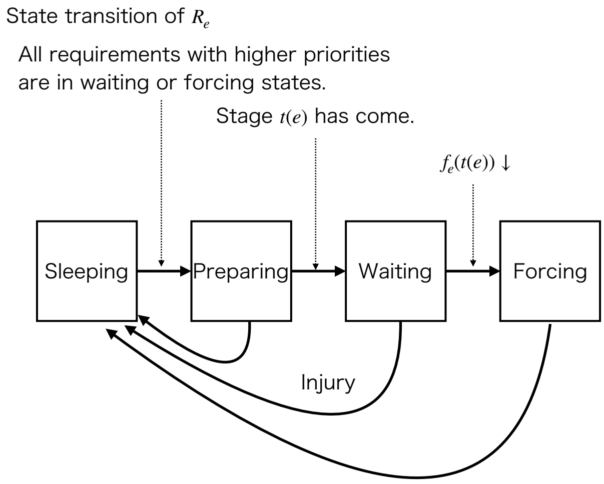

FIGURE 3. State transition

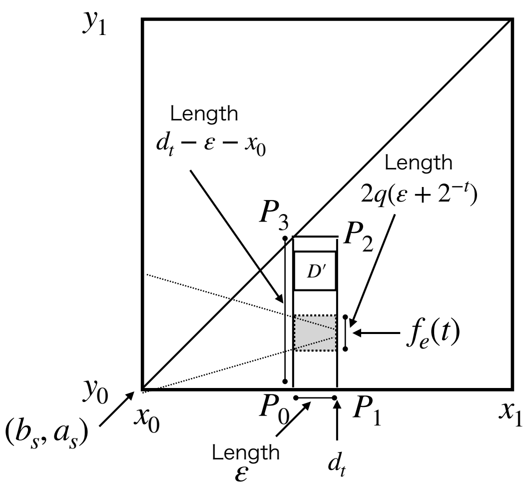

FIGURE 4. Local strategy

When the strategy is newly active (or at stage $s = 0$ in the local strategy), pick up a sufficiently large $t \in \omega$, which we will specify in (9) later. The strategy continues to be preparing for the stage $s \leq t$. Then, define (b_s, a_s) to be the bottom-left vertex of the forcing region and d_s to lie somewhere along the x-coodinate of the forcing region, say,

$$b_s = 0, \ a_s = 0, \ d_s = 1/2 \tag{6}$$

for all $s \leq t$.

Waiting state.

For the stage $s > t$, as long as $f_e(t)$ is not defined at stage s, the strategy is in the waiting state and define b_s, a_s, d_s by the equation (6). When running the combined strategy later, we need to define them differently.

Forcing state.

Suppose that $f_e(t)$ is defined exactly at stage u. For the stage $s \geq u$, the strategy is in the forcing state.

When the strategy is newly forcing at stage u, construct a new forcing region $D' = [x'_0, x'_1] \times [y'_0, y'_1]$ as Figure 4. Let $\varepsilon > 0$ be a sufficiently small rational.

To satisfy condition (I), let us focus on the rectangle $P_0P_1P_2P_3$ defined by

$$D'' = [d_t - \varepsilon, d_t] \times [y_0, y_0 + d_t - \varepsilon - x_0]. \tag{7}$$

Since the (old) forcing region D is a square, the distance between P_0 and (x_0, y_0) is the same as between P_0 and P_3. The region D'' is contained in the bottom-right triangle. Thus, if (β, α) is in the region D'', we have

$$\alpha - a_s \leq d_t - \varepsilon - x_0 \leq \beta - b_s$$

for all $s < u$, which implies condition (I) with $q = 1$ for all $s < u$.

To satisfy condition (5), it is sufficient for the new region D' to avoid the rectangle

$$D''' = [d_t - \varepsilon, d_t] \times [f_e(t) - q_e(\varepsilon + 2^{-t}), f_e(t) + q_e(\varepsilon + 2^{-t})]. \tag{8}$$

as long as the region is in the rectangle D'' defined in (7).

We can pick up such a square D' with the side length ε if the height of D'' defined in (7) is sufficiently longer than that of D''' defined (8), say,

$$2\varepsilon + 2q_e(\varepsilon + 2^{-t}) \leq d_t - \varepsilon - x_0,$$

which is equivalent to

$$\varepsilon \leq \frac{d_t - x_0 - 2q_e 2^{-t}}{2q_e + 3}. \tag{9}$$

Notice that we have determined $d_t - x_0 = 1/2$ before choosing t. Thus, at the beginning of the preparing state, we can choose t so that the right-hand side of (9) is positive. At the beginning of the forcing state, we choose $\varepsilon > 0$ so that the inequality (9) holds.

For stages $s \geq u$ (or when the strategy is in the forcing state), we define (b_s, a_s) and (d_s, a_s) within D', for instance,

$$b_s = x'_0, \;\; a_s = y'_0, \;\; d_s = x'_0.$$

Finally, let $\alpha = \lim_s a_s$ and $\beta = \lim_s b_s$. This is the end of construction.

Now, we verify some properties assuming f_e is total. By way of construction, $(a_s)_s$ and $(b_s)_s$ are nondecreasing computable sequence of rationals converging to α, β, respectively, thus α, β are left-c.e. We also have $(d_s)_s \in \mathrm{CS}(\beta)$.

We assert that condition (I) holds. Fix $s \in \omega$. If $s < u$ (or when the strategy is in the preparing or waiting state at stage s), we have already shown this. If $s \geq u$ (or when the strategy in the forcing state at stage s), we have $\alpha = a_s$. Thus, condition (I) holds.

We claim condition (II) does not hold with $c_t = f_e(t)$ and q_e fixed above. This is because, by condition (8),

$$|\alpha - f_e(t)| \geq q_e(\varepsilon + 2^{-t}) \geq q_e(|\beta - d_t| + 2^{-t}).$$

4.3. **Idea of combined strategy.** Condition (5) requires verifying all possible pairings of $(c_n)_n$ and q. Since we can not computably enumerate all computable sequences of rationals, we computably enumerate all partial computable functions. Let $((f_e, q_e))_e$ be a computable enumeration of all possible pairs of partial computable functions from ω to $\mathbb{Q}$ and positive rationals.

We will construct nondecreasing $(a_s)_s$ and $(b_s)_s$ for which condition (I) holds with $q = 1$. We will also construct $(d_s)_s$. More concretely, we will define a_s, b_s, and d_s at stage s. We set the requirement as follows:

$$R_e : (\exists t \in \omega)[f_e(t)\downarrow \Rightarrow |\alpha - f_e(t)| \geq q_e(|\beta - d_t| + 2^{-t})].$$

We further assume that if $f_e(t)$ is defined at stage s, then $s > t$. We employ the finite injury priority argument along the following priority of requirements:

$$R_0 > R_1 > R_2 > \cdots .$$

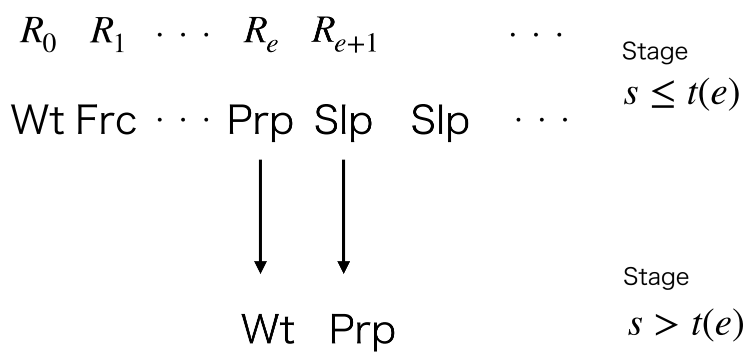

Figure 5. Transition of the preparing state

Each requirement is in one of the four states: sleeping, preparing, waiting, or forcing. Each preparing, waiting, or forcing R_e is associated with the witness number $t(e) \in \omega$. Each forcing requirement R_e is associated with the forcing region D_e. Notice that the witness number $t(e)$ and the region D_e are both dynamic; they change after an injury.

Only one requirement is in the preparing state at each stage s, usually denoted by $R_m = R_{m(s)}$. The requirements with higher priorities than R_m are in waiting or forcing states. The requirements with lower priorities than R_m are in sleeping state.

If R_e is in the preparing state at stage s, then it continues to be in the preparing state until the stage $t(e)$ unless an injury occurs. At stage $s = t(e) + 1$, R_e is in the waiting state and the next requirement R_{e+1} is in the preparing state as in Figure 5.

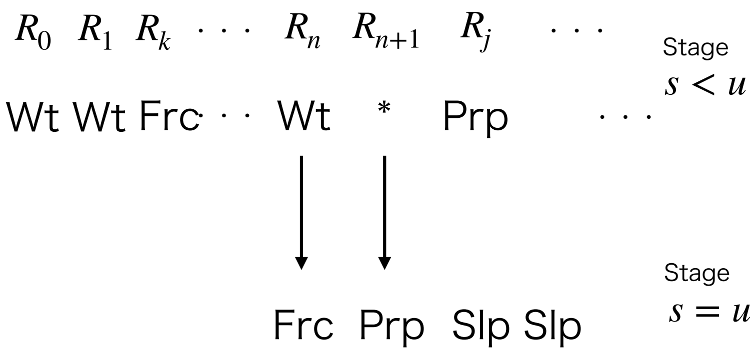

Figure 6. Injury

If n is the smallest index such tat $f_n(t(n))$ converges exactly at stage u, then R_n is in the forcing state, R_{n+1} is in the preparing state, and all requirements R_e for $e > n+1$ are in the sleeping state at stage u as in Figure 6. We say that R_n causes an injury in this case.

Let R_n be the requirement with the lowest priority among those in the forcing state and R_m be in the preparing state. By $D_n = [x_0^n, x_1^n] \times [y_0^n, y_1^n]$, we denote the region forced by R_n. Then, we define (b_s, a_s) as the lower-left corner of D_n. The real value d_s depends on which requirement R_m is in the preparing state. If $R_{n+1} = R_m$ is preparing, then $d_s = x_1^n$, which is the x-coordinate of the right side of the square D_n. When a requirement of a larger suffix enters preparing state (in other words, when m increases) in a later stage, say stage u, a value smaller than d_s is chosen as d_u. Figure 7 indicates locations that may be chosen as d_u depending on preparing requirements.

If $R_{n'}$ causes an injury, then a new forcing region will be created along the line $x = d_{t(n')}$. Then the candidates of d for lower requirements $R_{n'+1}, R_{n'+2}, \cdots$ are canceled. Afterwords, d for these requirements will be redefined inside the new forcing region $D_{n'}$.

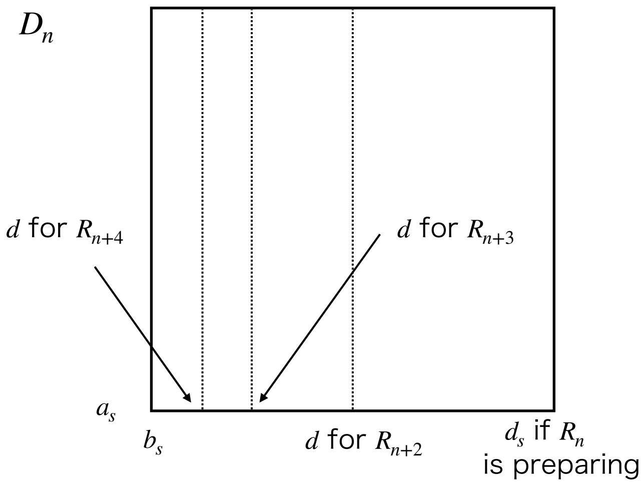

FIGURE 7. Definition of d_s

Notice that d for higher requirements $R_{n+1}, \ldots, R_{n'-1}$ are larger than $d_{t(n')}$. Thus, if $R_{n'}$ is injured by some of these requirements, then we can create a new forcing region on the right side of $D_{n'}$. This ensures that $(a_s)_s$ and $(b_s)_s$ are nondecreasing.

4.4. **Construction.** We give a concrete construction here.

Let $R_{n(s)}$ be the requirement with the lowest priority among those in the forcing state at stage s. If such an requirement does not exist, let $n(s) = -1$. Let $R_{m(s)}$ be the preparing requirement at stage s.

The initial forcing region is given by

$$D_{-1} = [0,1] \times [0,1].$$

For each stage s, we do the following five tasks.

(a) Update the states.
(b) Define D_n (if necessary).
(c) Define t (if necessary).
(d) Define (b_s, a_s).
(e) Define d_s.

(a) At stage $s = 0$, R_0 is in the preparing state and all other requirements are in the sleeping state.

At stage $s \geq 1$, first check whether an injury occurs. In other words, for each requirement R_e in the waiting state at stage $s-1$ associated with the witness number $t(e, s-1)$, check whether $f_e(t(e, s-1))$ is defined at stage s. If such requirements exist, then an injury occurs.

Suppose an injury occurs. Then, let $R_{n(s)}$ be the requirement with the highest priority among those requirements. At stage s, R_e is in the same state as stage $s-1$

for $e < n(s)$, $R_{n(s)}$ is the forcing state, $R_{n(s)+1}$ is in the preparing state, and all other requirements in the sleeping state as in Figure 6.

Suppose no injury occurs at stage s. Then, check whether the preparing requirement in the previous stage has done its job. Let $R_{m(s-1)}$ be the preparing requirement and $t(m(s-1))$ be its witness number. If $s \le t(m(s-1))$, then all requirements at stage s remain in the same states as stage $s-1$. If $s > t(m(s-1))$, then R_i is in the same state as stage $s-1$ for $i < m(s-1)$, $R_{m(s-1)}$ is in the waiting state, $R_{m(s)} = R_{m(s-1)+1}$ is in the preparing stage, and all other requirements with lower priorities are in the sleeping state as in Figure 5 with $e = m(s-1)$.

(b) If an injury occurs, we create a new forcing region. Suppose that the requirement $R_n = R_{n(s)}$ causes an injury. Then R_n has the lowest priority among all requirements in the forcing state. Let R_k be the forcing requirement with the next lowest priority. Note that $k < n$. If no such requirement exists, let $k = -1$. We create a new forcing region D_n for this requirement R_n in the forcing region $D_k = [x_0^k, x_1^k] \times [y_0^k, y_1^k]$. Let $t = t(n)$ be the witness number for R_n.

We consider the following two cases.

(b-1) The previous forcing region D_p is in the current D_k.

(b-2) No forcing region is created in the current D_k.

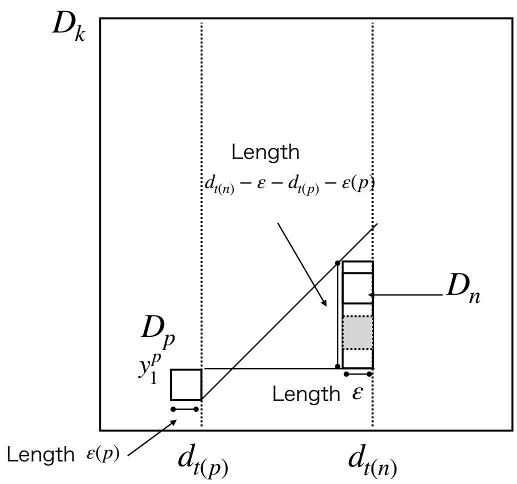

FIGURE 8. Construction of D_n

(b-1) At stage $s-1$, R_p is the requirement in the forcing state with the next highest priority to R_k. The D_p is the previous immediate child of D_k and may not be the narrowest forcing region in the previous stage $s-1$. Note that $k < n < p$. Let

$$D_p = [d_{t(p)} - \varepsilon(p), d_{t(p)}] \times [y_1^p - \varepsilon(p), y_1^p].$$

We choose a square D_n with side length ε such that

$$D_n \subseteq [d_t - \varepsilon, d_t] \times [y_1^p, y_1^p + L], \tag{10}$$

where

$$L = d_t - \varepsilon - d_{t(p)} - \varepsilon(p), \tag{11}$$

and D_n is disjoint from

$$[d_t - \varepsilon, d_t] \times [f_n(t) - q_n(\varepsilon + 2^{-t}), f_n(t) + q_n(\varepsilon + 2^{-t})] \tag{12}$$

as in Figure 8. We further assume that

$$\varepsilon < 2^{k-n-1}(x_1^k - x_0^k) \tag{13}$$

so that the existence of D_n can be proved by induction.
(b-2) In this case, the situation is similar to that in Figure 4. We choose a square D_k in a similar manner by regarding D_p as the point of the lower-left corner of D_n in Figure 8.

(c) Suppose that R_m is in the new preparing requirement at stage s. We define the witness number $t = t(m) > s$ for R_m such that

$$2^{n-m-1}(x_1^n - x_0^n) - 2q_m 2^{-t} > 0 \tag{14}$$

where $n = n(s)$. The witness number of each non-new preparing, waiting, and forcing requirement R_e remains the same as stage $s - 1$. Each sleeping requirement does not have the witness number.

(d) We define (b_s, a_s) as the the lower-left point of D_n forced by R_n where $n = n(s)$.

(e) Let $R_m = R_{m(s)}$, $R_n = R_{n(s)}$, and $D_n = [x_0^n, x_1^n] \times [y_0^n, y_1^n]$. We define

$$d_s = x_0^n + 2^{n-m+1}(x_1^n - x_0^n) \tag{15}$$

as in Figure 7.

Finally, let $\alpha = \lim_s a_s$ and $\beta = \lim_s b_s$. This is the end of construction.

4.5. **Verification.** We claim that we can find such D_n in (b). We give a proof for case (b-1). The following inequality is a sufficient condition:

$$2\varepsilon + 2q_n(\varepsilon + 2^{-t}) \leq L,$$

where L is the height of the possible region (10).

Now, $m(t(n))$, the preparing requirement at stage $t(n)$, is R_n; recall Figure 5. In other words,

$$n = m(t(n)). \tag{16}$$

The requirement with lowest priorities among those in the forcing state at stage $t(n)$ is R_k. Thus, the definition of d_s in equation (15) can be rewritten by

$$d_{t(n)} = x_0^k + 2^{k-m(t(n))+1}(x_1^k - x_0^k) = x_0^k + 2^{k-n}(x_1^k - x_0^k),$$

Hence, we can evaluate L by (11) from below as follows:

$$\begin{aligned}L &= d_{t(n)} - d_{t(p)} - \varepsilon - \varepsilon(p)\\ &> 2^{k-n}(x_1^k - x_0^k) - \varepsilon - 2^{k-n-2}(x_1^k - x_0^k)\\ &> 2^{k-n-1}(x_1^k - x_0^k) - \varepsilon.\end{aligned}$$

Here, we used (13) and $n+1 \leq p$ to deduce

$$\varepsilon(p) < 2^{k-p-1}(x_1^k - x_0^k) \leq 2^{k-n-2}(x_1^k - x_0^k).$$

Thus, the following inequality is a sufficient condition of ε:

$$\varepsilon \leq \frac{2^{k-n-1}(x_1^k - x_0^k) - 2q_n 2^{-t}}{2q_n + 3}.$$

We can find such a positive rational ε by inequality (14)

Lower-left coordinates of the forcing regions are always increasing. Thus, $(a_s)_s$ and $(b_s)_s$ are nondecreasing and their limits, α and β, are left-c.e. By construction, each requirement R_e will be met.

Finally, we claim that $\alpha - a_s \leq \beta - b_s$ for each $s \in \omega$. The point (b_s, a_s) moves only when the new forcing region D_n is created at stage s. By construction in (b), in particular by (10), we have $a_s - a_{s-1} \leq b_s - b_{s-1}$ for such stages s. This implies the claim that $\alpha - a_s \leq \beta - b_s$ for each $s \in \omega$.

This is the end of the proof.

5. Second counterexample

In this section, we prove that condition (III) is strictly stronger than (I). The meaning and structure of this section are similar to the previous one, but the proof here is simpler.

5.1. **Claim.**

Theorem 5.1. *There exist $\alpha, \beta \in$ **CA** such that condition (I) holds but condition (III) does not. Additionally, we can impose $\alpha \in \mathbb{Q}$.*

We define $a_s = 0$ for all $s \in \omega$ and $\alpha = 0$. Thus, condition (I) clearly holds as long as $\beta \in \mathbf{CA}$.

The precise statement of condition (III) is

$$(\forall (a_s))(\exists (b_s))(\exists q)(\forall s)[|\alpha - a_s| < q(|\beta - b_s| + 2^{-s})].$$

Considering $\alpha = 0$, we require the negation of this condition, that is,

$$(\exists (c_n)_n \in \mathrm{CS}(0))(\forall (d_n)_n \in \mathrm{CS}(\beta))(\forall q \in \omega)(\exists t \in \omega)[|c_t| \geq q(|\beta - d_t| + 2^{-t})].$$

We define

$$c_t = 2^{-t/2}.$$

Then, the condition can be written as

$$(\forall (d_n)_n \in \mathrm{CS}(\beta))(\forall q \in \omega)(\exists t \in \omega)[|\beta - d_t| \leq r_e(t)],$$

where

$$r_e(t) = (q_e^{-1} - 2^{-t/2})2^{-t/2}.$$

Notice that, if $t > 2\log_2 q_e$, then $q_e^{-1} - 2^{-t/2} > 0$ and $r_e(t) > 0$.

We computably enumerate all pairs (f_e, q_e) of partial computable functions from ω to $\mathbb{Q}$ and positive rationals. For each $e \in \omega$, we set a requirement R_e by

$$R_e : \ (f_e(s))_s \in \mathrm{CS}(\beta) \implies (\exists t > 2\log_2 q_e)[|\beta - f_e(t)| \leq r_e(t)].$$

These requirements are sufficient to ensure the condition above. We set the priority of the requirements as

$$R_0 > R_1 > R_2 > \cdots .$$

5.2. **Local strategy.** We explain a strategy for a single requirement R_e for a fixed e. Recall that $f_e : \omega \to \mathbb{Q}$ is a partial computable function, which may represent $(d_n)_n \in \mathrm{CS}(\beta)$ and q_e is a positive rational.

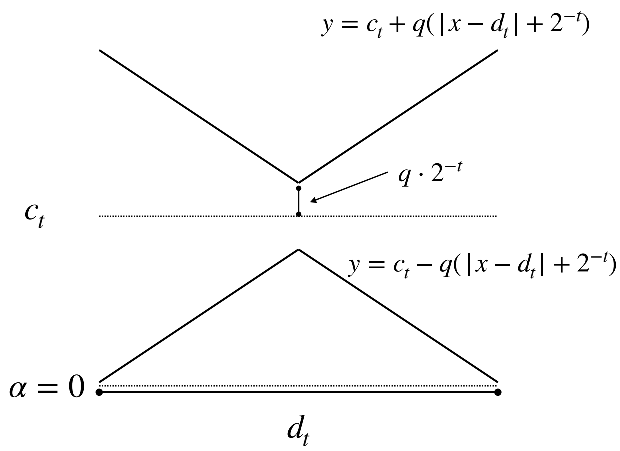

Figure 9. Possible (β, α)

We want to enforce that (β, α) is not between the two poly-lines $y = c_t \pm q(|x - d_t| + 2^{-t})$ for some t as in Figure 9.

Considering $a_s = 0$ for all s, we force β to be close to $f_e(t)$ when $f_e(t)$ is defined. To respect requirements with higher priorities, the forcing interval should be contained in the previous forcing interval, which is $[0, 1]$ in the single strategy. Thus, we need to wait t such that $f_e(t)$ is defined and it is in this interval.

This strategy operates in two states:

(i) Waiting state.
(ii) Forcing state.

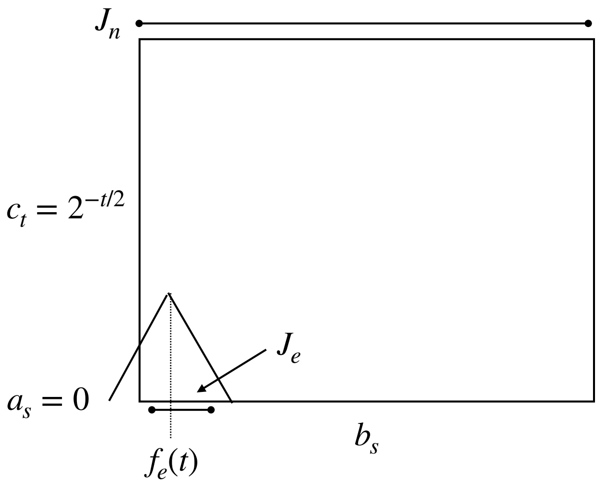

FIGURE 10. Construction of J_e

In the single strategy, let $n = -1$ and $J_n = [0, 1]$ be the initial forcing closed interval for β. By J_e, we denote the interval forced by requirement R_e.

Later, in the combined strategy, we use n as the index of the forcing requirement with the lowest priority. If such requirement does not exist, $n = -1$ for convenience.

At stage s, search $t \le s$ such that

$$t > 2 \log_2 q_e,\ \ f_e(t)[s] \downarrow \in J_n^o, \tag{17}$$

where A^o denotes the interior of the set $A \subseteq \mathbb{R}$. If such t is found, the strategy transitions from the waiting state to forcing state. Note that this t is similar to the witness number in the previous section's proof, but it cannot be fixed in advance. To respect the previous forcing interval J_n, the new forcing interval should be contained in it. To do this, we need to find t such that $f_e(t) \in J_n^o$. Since $(f_e(s))_s$ may converges so slowly, this t may be really large.

If such t is found, we define a new forcing closed interval J_e with rational endpoints satisfying

$$J_e \subset J_n^o \cap [f_e(t) - r_e(t),\ f_e(t) + r_e(t)],\ 0 < |J_e| \le \frac{1}{2}|J_n|.$$

The b_s is the midpoint of the narrowest forcing interval. Thus, b_s is the midpoint of J_{-1}, say $1/2$, initially, and b_s is the midpoint of J_e if defined. The limit of $(b_s)_s$ is defined to be β.

We are going to show that the requirement R_e is satisfied. Suppose that $(f_e(s))_s \in \mathrm{CS}(\beta)$. Since $\beta \in J_n^o$ regardless of whether J_e is defined or not, we have $f_e(t) \in J_n^o$ for a sufficiently large t. Thus, condition (17) is satisfiend and J_e is eventually defined. Hence, R_e is satisfied.

5.3. **Construction of combined strategy.** Now we combine the previous single strategies for all $e \in \omega$ to make all requirements R_e satisfied. Each requirement is in the waiting or forcing state. If a requirement transitions from the waiting state to the forcing state, then all requirements with lower priorities reset to the waiting state.

Each forcing requirement R_e is associated with the forcing closed interval $J_{e,s}$, which changes as stage s goes. For convenience, let $J_{-1,s} = [0,1]$ for all $s \in \omega$. Let $n(e,s)$ denote the index of the forcing requirement with the lowest priority among those with higher priority than R_e at stage s. For example, if requirements R_1, R_3 are in the forcing state and R_0, R_2 are in the waiting state at stage s, then $n(0,s) = -1$, $n(2,s) = 1$ and $n(4,s) = 3$.

We will define b_s at stage s.

For each stage s, we do the following three tasks.

(a) Update the states.
(b) Define $J_{n,s}$.
(c) Define b_s.

(a) For each $e \leq s$ such that R_e is in the waiting state at $s-1$, search $t \leq s$ such that

$$t > 2\log_2 q_e, \ \ f_e(t)[s] \downarrow \in J^o_{n(e,s-1),s-1}, \tag{18}$$

where A^o is the interior of the set $A \subseteq \mathbb{R}$. If such R_e exists, then an injury occurs.

If an injury occurs, let R_k be the requirement with the highest priority among those requirements. At stage s, each R_e for $e < k$ is in the same state as stage $s-1$, R_k is in the forcing state, and R_e for $e > k$ is in the waiting state.

If no injury occurs, all requirements are in the same state as stage $s-1$.

(b) For each waiting requirement R_e, $J_{e,s}$ is undefined.

If no injury occurs, all forcing intervals $J_{e,s}$ of forcing requirement R_e are the same as $J_{e,s-1}$.

If an injury occurs, we define a new forcing interval. Let R_k be the new forcing requirement and let R_n be the forcing requirement with the next lowest priority. Note that $n = n(k,s)$. For any forcing requirement R_e other than R_k, let $J_{e,s} = J_{e,s-1}$.

We define a new forcing closed interval $J_{k,s}$ with rational endpoints satisfying

$$J_{k,s} \subseteq J^o_{n,s} \cap [f_k(t) - r_k(t), \ f_k(t) + r_k(t)], \ 0 < |J_{k,s}| \leq \frac{1}{2}|J_{n,s}|,$$

where t is the witness found for R_k in (a).

(c) Let R_e be the forcing requirement with the lowest priority. Let b_s be the midpoint of $J_{e,s}$.

Finally, let β be the limit of $(b_s)_s$. This is the end of the construction.

5.4. **Verification.** We claim that for each $e \in \omega$, $J_{e,s}$ stabilizes to a closed interval, which is denoted by J_e, or is undefined for all sufficiently large s. Notice that each requirement R_e is injured only by those with higher priority, which are finite. Thus, the claim can be proved by induction on e. By the same reason, $n(e, s)$ converges to, say, $n(e)$.

Next, we claim that $(b_s)_s$ converges. For each e such that J_e is defined, b_s is in the interval J_e for all sufficiently large s. Since the lengths of the forcing intervals J_e decreases to 0, the sequence $(b_s)_s$ converges.

Since $\alpha = 0$ and $\beta \in \mathbf{CA}$, condition (I) clearly holds.

We claim that, for each $e \in \omega$, the requirement R_e is satisfied. Suppose that $(f_e(s))_s \in \mathrm{CS}(\beta)$. Since $\beta \in J^o_{n(e)}$, $f_e(s) \in J^o_{n(e)}$ for all sufficiently large s. Thus, condition (18) holds for all sufficiently large t. Therefore, R_e will eventually enter the forcing state. By way of construction of $J_{e,s}$, the requirement R_e is satisfied. This concludes the proof.

6. Modification of error terms

We have observed that Solovay reducibility for c.a. reals is less robust than that for left-c.e. reals. As a final remark, we demonstrate that by adjusting the error term, robustness can be achieved concerning the choice of quantifiers.

Let R be the set of all computable sequences of positive rationals converging to 0. For $\alpha, \beta \in \mathbf{CA}$, $(a_s)_s \in \mathrm{CS}(\alpha)$, and $(b_s)_s \in \mathrm{CS}(\beta)$, let P^E_0 be the relations defined by the following:

$$P^E_0 :(\exists (r_s)_s \in R)(\exists q \in \omega)(\forall s \in \omega)[|\alpha - a_s| < q|\beta - b_s| + r_s].$$

We then consider the following conditions for $\alpha, \beta \in \mathbf{CA}$:

(E-I) $(\exists (a_s))(\exists (b_s))P^E_0$,
(E-II) $(\forall (b_s))(\exists (a_s))P^E_0$,
(E-III) $(\forall (a_s))(\exists (b_s))P^E_0$,
(E-IV) $(\exists (b_s))(\forall (a_s))P^E_0$,
(E-V) $(\exists (a_s))(\forall (b_s))P^E_0$,
(E-VI) $(\forall (a_s))(\forall (b_s))P^E_0$.

We again assume $(a_s)_s \in \mathrm{CS}(\alpha)$ and $(b_s)_s \in \mathrm{CS}(\beta)$ when considering these quantifiers.

Proposition 6.1. *For $\alpha, \beta \in \mathbf{CA}$, condition (E-I) is equivalent to $\alpha \leq_S \beta$.*

Proof. If $\alpha \leq_S \beta$ via $(a_s)_s$ and $(b_s)_s$, condition (E-I) holds via $(a_s)_s$, $(b_s)_s$ and $r_s = 2^{-s}$.

Suppose (E-I) holds for some $(a_s)_s \in \mathrm{CS}(\alpha)$, $(b_s)_s \in \mathrm{CS}(\beta)$, and $(r_s)_s \in R$. Since $r_s \to 0$ as $s \to \infty$, there exists a computable increasing function $g(s)$ such that $r_{g(s)} \leq 2^{-s}$. Thus, we obtain $\alpha \leq_S \beta$ via $(a_{g(s)})s$ and $(b_{g(s)})_s$. ☐

Theorem 6.2. *Condition (E-I) implies condition (E-VI). Thus, all six conditions, from (E-I) to (E-VI), are mutually equivalent.*

Proof. Assume condition (E-I) holds for a pair $(a_n)_n \in \mathrm{CS}(\alpha)$ and $(b_n)_n \in \mathrm{CS}(\beta)$. Consider $(c_n)_n \in \mathrm{CS}(\alpha)$ and $(d_n)_n \in \mathrm{CS}(\beta)$. Then,

$$\begin{aligned} |\alpha - c_n| &\leq |\alpha - a_n| + |a_n - c_n| \\ &< q|\beta - b_n| + r_n + |a_n - c_n| \\ &\leq q|\beta - d_n| + r_n + |a_n - c_n| + q|b_n - d_n|. \end{aligned}$$

Since $r_n + |a_n - c_n| + q|b_n - d_n|$ converges to 0, P_0^E holds with this modified error term. ☐

This modification improves the robustness of the definition of Solovay reducibility.

Remark 6.3. Proposition 6.1 and Theorem 6.2 provide further evidence that Solovay reducibility is a natural notion. We have shown that conditions (II) and (III) differ from Solovay reducibility, but we do not claim that they are unnatural.

Acknowledgement

This work was supported by JSPS KAKENHI Grant Numbers JP21K03340, JP22K03408, and JP21K18585. This work was also supported by the Research Institute for Mathematical Sciences, an International Joint Usage/Research Center located in Kyoto University.

References

1. Rod G. Downey, Denis R. Hirschfeldt, and Geoff LaForte, *Randomness and reducibility*, J. Comput. System Sci. **68** (2004), no. 1, 96–114. MR 2030512
2. Rodney G. Downey and Denis R. Hirschfeldt, *Algorithmic randomness and complexity*, Theory and Applications of Computability, Springer, New York, 2010. MR 2732288
3. Antoní n Kučera and Theodore A. Slaman, *Randomness and recursive enumerability*, SIAM J. Comput. **31** (2001), no. 1, 199–211. MR 1857396
4. Joseph S. Miller, *On Work of Barmpalias and Lewis-Pye: A Derivation on the D.C.E. Reals*, Computability and Complexity - Essays Dedicated to Rodney G. Downey on the Occasion of His 60th Birthdayedicated to Rodney G. Downey on the Occasion of His 60th Birthday (Adam R. Day, Michael R. Fellows, Noam Greenberg, Bakhadyr Khoussainov, Alexander G. Melnikov, and Frances A. Rosamond, eds.), Lecture Notes in Computer Science, vol. 10010, Springer, 2017, pp. 644–659.
5. Robert Rettinger and Xizhong Zheng, *Solovay reducibility on d-c.e real numbers*, Computing and Combinatorics: 11th Annual International Conference, COCOON 2005, Kunming, China, August 16-19, 2005, Proceedings, vol. 3595, Springer, 2005, pp. 359–368.
6. R. Solovay, *Draft of paper (or series of papers) on chaitin's work.*, Unpublished notes, May 1975. 215 pages., 1975.

7. Xizhong Zheng and Robert Rettinger, *On the extensions of Solovay-reducibility*, Computing and Combinatorics: 10th Annual International Conference, COCOON 2004, Jeju Island, Korea, August 17-20, 2004, Proceedings (K.-Y. Chwa and J.I. Munro, eds.), vol. 3106, Springer, 2004, pp. 360–369.

(M. Kumabe) The Open University of Japan
Email address: kumabe@ouj.ac.jp

(K. Miyabe) Meiji University, Japan
Email address: research@kenshi.miyabe.name

(T. Suzuki) Tokyo Metropolitan University, Japan
Email address: toshio-suzuki@tmu.ac.jp