SEQUENTIAL COMPUTABILITY OF A FUNCTION -LIMITING RECURSION VERSUS EFFECTIVE UNIFORMITY- TO THE MEMORY OF MARIAN BOYKAN POUR-EL

MARIKO YASUGI* AND MASAKO WASHIHARA**

Received January 23, 2010

Abstract. The major objective of this article is a refinement of treatments of the mutual relationship between two notions of "sequential computability" of a function which is possibly Euclidean-discontinuous, one using limiting recursion and one using effective uniformity. We also speculate on these methods from a mathematician's viewpoint.

1. Introduction

The objective of this article is to distill the general situation in which two methods of computing some Euclidean-discontinuous functions become equivalent. Those are the methods developed respectively on "effective uniformity" and on "limiting recursion." In so doing, we speculate on the two notions of "sequential computability."

The domain of our discourse will be the real line or a subinterval of the real line as well as some (possibly Euclidean-discontinuous) functions on it.

It has been an old practice to review mathematics from the algorithmic viewpoints. The underlying method is the elementary theory of recursive functions (or its equivalents). On the continuum, a computable object is approximated by a recursive sequence of rational numbers or a recursive sequence from a discrete structure with a recursive modulus of convergence (effective approximation).

Investigation of computability on the continuum is based on the "computable sequence of reals." Computability of real functions was originally defined for continuous functions (cf. Chapter 0 in [4], for example). A continuous real function is called computable if it maps any computable sequence of real numbers to a computable sequence (sequential computability), and it has a recursive modulus of continuity (effective continuity). The sequential computability is required for the following reason. In order to claim that a function be computable, one must have a general algorithm to compute the value of that function for any computable real number. In order to secure it, it is known to be sufficient to assume the sequential computability.

One might expect that one could compute the values of a function without the assumption of continuity, but it is not so. A simple function such as the floor function $[x]$ (called also the Gaußian function), which jumps at each integer but is continuous (constant) on the interval of two adjacent integers, does not preserve sequential computability ([10]).

Such a problem has been discussed in [10], [8], [11]. In fact, we, human beings, easily compute such a function at any (computable) point. We know that, for any integer n, $[x]$ satisfies the requirements for the computability of a continuous function on the interval

2000 *Mathematics Subject Classification.* 00A35, 03Dxx .

Key words and phrases. computability in analysis, limiting recursion, effective uniformity, sequential computability .

The first author has been supported in part by JSPS Grant-in-Aid No. 20540143.

$[n, n+1)$ (and the value is n). It is all too easy! There are, however, many Euclidean-discontinuous functions which are far more complex than $[x]$ and still allow us to somehow "compute" their values.

We wished to express such an intellectual activity (in computing the values of a discontinuous function) of the human mind in a mathematical language, and have proposed two such treatments: one expressing such a computation in terms of "limiting recursive functions" of natural numbers ([10], [8]), and one in terms of "changing topology" of the domain of a function, thus regarding a Euclidean-discontinuous function like $[x]$ as continuous in the new topology so that we can conceive the computability of a function as that of a continuous function ([6]). Each of them can be interpreted as expressing a certain human intellectual activity for a same purpose from different viewpoints.

Both methods have been well studied and applied to many examples of Euclidean-discontinuous functions ([7], [10], [12], [14], [16]). Analyzing these individual treatments, we have pinpointed a *general framework* under which two notions of sequential computability "nearly" coincide. The framework will be introduced as the assumption $[\mathcal{A}]$ and the condition $[\mathcal{C}]$ in Section 3.

It is notable that "near equivalence" of the two notions holds notwithstanding that the two approaches are methodologically quite different. This has forced the authors to speculate on the meanings of the two approaches. As for the limiting recursion method, we have already discussed its significance and problems in [8]. Here we will put emphasis on the contrast between the two methods.

To us, the method of uniform spaces seems most natural and intuitive. It represents the freedom and flexibility that our mind enjoys.

We present very briefly in Section 2 some basics of computability on the continuum. No mathematical details will be supplied except for what is necessary to our present purpose. For basics of computability in analysis, we refer the reader to [4] and [15].

Our interest lies in the real function which is Euclidean-discontinuous but is fairly tame so that one can attribute to it some kind of computability property. We will hence set up a framework to meet our purpose in Section 3. Two (extended) notions of "sequential computability" of a function in our framework are then formulated in Section 4.

Our main results (Theorems 1 and 2), expressing a close relationship of the two notions of sequential computability in our general framework, are proved in Section 5 under a set of premises $(\mathcal{D})$ on a function.

An example of computation in the respective method according to our framework is explained in Section 6. The article is concluded with a speculation on limiting recursion versus effective uniformity in Section 7.

A similar discussion is also seen in [9]. We have also worked on a sequence of uniformities and its limit; some mathematical results as well as the significance of such a theory are seen in [11].

2. Preliminaries

We will list some of the basic notions and notations which are just necessary to our discussion. In the following, $m, n, p, q, k, \cdots$ will denote positive (or non-negative) integers.

Definition 2.1. (Computable real sequence) (i) A sequence of real numbers $\{x_m\}$ is called E-*computable* (computable in the Euclidean topology) if the following hold ([4]).

(1) There is a recursive (double) sequence of rational numbers $\{r_{mn}\}$ which approximates $\{x_m\}$.

(2) There is a recursive modulus of convergence of $\{r_{mn}\}$ to $\{x_m\}$, say β, that is,

$$n \geq \beta(m,p) \rightarrow |x_m - r_{mn}| < \frac{1}{2^p}.$$

In such a case, and in any similar situation, we say that $\{r_{mn}\}$ *effectively approximates* (converges to) $\{x_m\}$.

(ii) A number-theoretic function η is called after Gold [2] *limiting recursive* if it is defined to be the limit of a recursive function, that is, there is a recursive function h satisfying $\eta(p) = \lim_n h(p,n)$, presuming that the limit exists. (In fact, a function which is recursive in a limiting recursive function will also be called limiting recursive.)

(iii) If in (2) of (i) above the recursive β be replaced by a limiting recursive η, then we say that $\{x_m\}$ is *weakly* E-*computable* (by $\{r_{mn}\}$ and η).

We will define the effective uniform topology on an arbitrary non-empty set X, although the universe of our discourse will be the set of real numbers $\mathbf{R}$ or its subintervals. (We have employed the definition of (classical) uniformity in [3].)

Definition 2.2. (Effective uniformity:[6]) $\mathcal{U} = \{U_n\}$ is called an *effective uniformity* on X if U_n is a map from X to the powerset of X for each n, and there are recursive functions $\alpha_1, \alpha_2, \alpha_3$ which satisfy the following.

$$\forall x \in X. \cap_n U_n(x) = \{x\}.$$
$$\forall n, m \forall x \in X. U_{\alpha_1(n,m)}(x) \subset U_n(x) \cap U_m(x).$$
$$\forall n \forall x, y \in X. x \in U_{\alpha_2(n)}(y) \rightarrow y \in U_n(x).$$
$$\forall n \forall x, y, z \in X. x \in U_{\alpha_3(n)}(y) \wedge y \in U_{\alpha_3(n)}(z) \rightarrow x \in U_n(z).$$

It is known that $\langle X, \{U_n\}\rangle$ is a (uniform) topological space with $\{U_n(x)\}$ as the system of fundamental neighborhoods. We will call $\langle X, \{U_n\}\rangle$ an *effective uniform space*.

Definition 2.3. (Effective $\mathcal{U}$-convergence:[6]) A double sequence $\{r_{mn}\}$ from X is said to *effectively* $\mathcal{U}$-*converge* to a sequence $\{x_m\}$ if there is a recursive function γ satisfying $\forall m \forall n \forall k \geq \gamma(m,n). r_{mk} \in U_n(x_m)$. We also say that $\{x_m\}$ is the *effective* $\mathcal{U}$-*limit* of $\{r_{mn}\}$ with modulus of convergence γ.

Note If in Definition 2.3 γ be replaced by a limiting recursive function η, then we say that $\{r_{mn}\}$ *weakly* $\mathcal{U}$-*converges* to $\{x_m\}$.

Definition 2.4. ($\mathcal{U}$-computable sequences:[6]) (1) A family of sequences from X, say $\mathcal{S}$, is called a $\mathcal{U}$-computability structure if it is closed under recursive re-enumeration and $\mathcal{U}$-effective convergence.

(2) A sequence in $\mathcal{S}$ is called $\mathcal{U}$-computable.

(3) An element x of X is called $\mathcal{U}$-computable if $\{x, x, \cdots\}$ is in $\mathcal{S}$.

3. Framework

We will here set up a framework in order to attain our purpose. We will first place an overall assumption.

Assumption $[\mathcal{A}]$ We work in an effective uniform space on the set of real numbers $\mathbf{R}$, $\mathcal{U} = \langle \mathbf{R}, \{U_n\}\rangle$, and assume the following.

$\mathcal{A}$-1 A recursive sequence of rational numbers is $\mathcal{U}$-computable.

$\mathcal{A}$-2 E-computable numbers and $\mathcal{U}$-computable numbers coincide.

$\mathcal{A}$-3 Every $\mathcal{U}$-computable sequence is E-computable.

Note It should be noted that the converse of $\mathcal{A}$-3 is *not* assumed. In fact, the converse does not hold in any significant case, and that is the essential point of changing the topology.

We further assume a condition on $\mathcal{U}$, denoted by $[\mathcal{C}]$.

Condition $[\mathcal{C}]$ on $\mathcal{U}$ Given an E-computable sequence $\{x_m\}$, there is a $\mathcal{U}$- computable sequence $\{z_{mp}\}$ and a limiting recursive function ν such that $\{x_m\}$ is *weakly $\mathcal{U}$-computable* by $\{z_{mp}\}$ and ν, that is, $\{z_{mp}\}$ weakly $\mathcal{U}$-converges to $\{x_m\}$ (cf. Definition 2.3), or

$$\forall m, n \forall p \geq \nu(m,n). z_{mp} \in U_n(x_m). \tag{1}$$

We may say that $\{z_{mp}\}$ and ν are associated with $\{x_m\}$.

Note We have employed the term "weakly $\mathcal{U}$-computable" according to [14], in which the term is used in the context of "weakly Fine-computable."

Proposition 3.1. If $\{x_m\}$ is $\mathcal{U}$-computable, then ν can be recursive, since $z_{mp} = x_m$ will do.

Definition 3.1. (Framework) The *framework* of our subsequent study of real functions consists of $[\mathcal{A}]$ and $[\mathcal{C}]$.

Note (i) The condition $[\mathcal{C}]$ signifies that, although an E-computable sequence may not be $\mathcal{U}$-computable, it is $\mathcal{U}$-computable in a weak sense.

(ii) All the uniform spaces on $\mathbf{R}$ we have dealt with satisfy the assumption $[\mathcal{A}]$ and the condition $[\mathcal{C}]$ ([6], [7], [11], [12], [14]), and hence they represent natural requirements.

4. Two notions of sequential computability of a function

We here define two notions of sequential computability of a real function similarly to [12]. Although the definitions are stated for a function whose domain is the whole real line, the definitions can be easily modified to any interval with computable end-points. We will henceforth assume the Framework in Definition 3.1.

Definition 4.1. (Sequential computability of a function) (i) ($\mathcal{L}$-sequential computability) A real function f is called *$\mathcal{L}$-sequentially computable* if, for any E-computable sequence of real numbers $\{x_m\}$, which is weakly $\mathcal{U}$-computable by $\{z_{mp}\}$ and ν (cf. $[\mathcal{C}]$), the sequence of function values $\{f(x_m)\}$ is weakly E-computable (cf. (iii) of Definition 2.1) with a recursive sequence of rational numbers $\{s_{mn}\}$ and a function η which is recursive in ν.

(ii) ($\mathcal{U}$-sequential computability) f is called *$\mathcal{U}$-sequentially computable* if, for any $\mathcal{U}$-computable sequence of real numbers $\{x_m\}$, the sequence of function values $\{f(x_m)\}$ is E-computable.

Note In fact, $\mathcal{L}$-sequential computability of a function f should be defined independent of any uniformity. With each concrete example we have worked on, it was so defined, namely, a limiting modulus of convergence is defined independent of any uniformity, and later some conditions corresponding to those in the framework have been demonstrated. In an abstract setting, however, we must set those conditions as assumptions.

5. Mutual relationship of the two notions

We will prove the "near" equivalence of two notions of sequential computability of a real function as has been defined in Definition 4.1 in the general framework of Section 3. The proof is a generalized version of the corresponding ones in [12] and [14].

Theorem 1. (From $\mathcal{L}$-sequential computability to $\mathcal{U}$-sequential computability) If f is $\mathcal{L}$-sequentially computable, then f is $\mathcal{U}$-sequentially computable.

Proof Suppose f is $\mathcal{L}$-sequentially computable, and let $\{x_m\}$ be $\mathcal{U}$-computable. By $[\mathcal{A}]$, $\{x_m\}$ is E-computable, and hence, by $[\mathcal{C}]$, a limiting recursive ν is associated with it. So, by $\mathcal{L}$-sequential computability, there is a recursive sequence of rational numbers $\{t_{mp}\}$ and a function η which is recursive in ν satisfying

$$\forall m, p \forall q \geq \eta(m,p).|f(x_m) - t_{mq}| < \frac{1}{2^p}.$$

By virtue of Proposition 3.1, one can take a recursive ν for a $\mathcal{U}$-computable sequence $\{x_m\}$, and so we can take a recursive η so that $\{f(x_m)\}$ is E-computable by $\{t_{mq}\}$ and η, and hence f is $\mathcal{U}$-sequentially computable.

Note Notice that for Theorem 1 we do not need to assume any kind of continuity on the function f. The converse of Theorem 1 will be proved under "effective $\mathcal{U}$-continuity" as well as some supplementary conditions on the space. Effective $\mathcal{U}$-continuity can be found also in [12] and [14].

Definition 5.1. (Effective $\mathcal{U}$-continuity) f is called effectively $\mathcal{U}$-continuous if there is a $\mathcal{U}$-computable sequence $\{e_i\}$ and a recursive function γ which satisfy the following for each p.

$$\cup_i U_{\gamma(i,p)}(e_i) = \mathbf{R}; \quad \forall i, x.x \in U_{\gamma(i,p)}(e_i) \to |f(x) - f(e_i)| < \frac{1}{2^p}. \tag{2}$$

Condition $[\mathcal{D}]$ on a function f:

$\mathcal{D}$-1: f is effectively $\mathcal{U}$-continuous with $\{e_i\}$ and γ as in Definition 5.1.

$\mathcal{D}$-2: For each E-computable sequence $\{x_m\}$, with which ν is associated, there is a function ι, which is recursive in ν and satisfies, for each p,

$$\forall m.x_m \in U_{\gamma(\iota(m,p),p)}(e_{\iota(m,p)}).$$

$\mathcal{D}$-3: For each E-computable sequence $\{x_m\}$ with ν as above , there is a function ε which is recursive in ν and satisfies

$$\forall i, m, n.x_m \in U_n(e_i) \to U_{\varepsilon(m,i,n)}(x_m) \subset U_n(e_i).$$

Theorem 2. (From $\mathcal{U}$-sequential computability to $\mathcal{L}$-sequential computability) Assume that the condition $[\mathcal{D}]$ holds for a function f. In particular, f is effectively $\mathcal{U}$-continuous (cf. $\mathcal{D}$-1). If f is $\mathcal{U}$-sequentially computable, then f is $\mathcal{L}$-sequentially computable.

Proof Suppose f is $\mathcal{U}$-sequentially computable.

Let $\{x_m\}$ be E-computable, and let p be a positive integer.

(i) By $[\mathcal{C}]$, there are a $\mathcal{U}$-computable sequence $\{z_{mq}\}$ and a limiting recursive function ν such that

$$\forall m, n \forall q \geq \nu(m,n).z_{mq} \in U_n(x_m)$$

Since f is $\mathcal{U}$-sequentially computable and $\{z_{mp}\}$ is $\mathcal{U}$-computable, $\{f(z_{mp})\}$ is E-computable. From this follows:

(ii) There are a recursive sequence of rational numbers $\{s_{mql}\}$ and a recursive function β so that

$$\forall n, m, q \forall l \geq \beta(m,q,n).|f(z_{mq}) - s_{mql}| < \frac{1}{2^n}$$

(iii) Put $t_{mq} := s_{mq\beta(m,q,q)}$.

(iv) By $\mathcal{D}$-2, there is an ι, recursive in ν, such that $x_m \in U_{\gamma(i,p+2)}(e_i)$ holds, where $i = \iota(m, p+2)$.

(v) From (iv) and $\mathcal{D}$-3, there is a function ε which is recursive in ν so that with $n = \gamma(i, p+2)$,

$$U_{\varepsilon(m,i,\gamma(i,p+2))}(x_m) \subset U_{\gamma(i,p+2)}(e_i).$$

For short, put $\varepsilon_0(m,i,p) = \varepsilon(m,i,\gamma(i,p+2))$.
(vi) In (i), put $n = \varepsilon_0(m,i,p)$ to obtain

$$q \geq \nu(m, \varepsilon_0(m,i,p)) \to z_{mq} \in U_{\varepsilon_0(m,i,p)}(x_m).$$

(vii) From (v) and (vi), we obtain

$$q \geq \nu(m, \varepsilon_0(m,i,p)) \to z_{mq} \in U_{\gamma(i,p+2)}(e_i).$$

(viii) From $\mathcal{D}$-1 with $x = z_{mq}$ and $p = p+2$ in the formula (2) of Definition 5.1, we obtain

$$z_{mq} \in U_{\gamma(i,p+2)}(e_i) \to |f(z_{mq}) - f(e_i)| < \frac{1}{2^{p+2}}.$$

(ix) From (vii) and (viii), follows

$$q \geq \nu(m, \varepsilon_0(m,i,p)) \to |f(z_{mq}) - f(e_i)| < \frac{1}{2^{p+2}}.$$

(x) In $\mathcal{D}$-1, put $x = x_m$ and $p = p+2$. Since by (iv) $x_m \in U_{\gamma(i,p+2)}(e_i)$, it follows

$$|f(x_m) - f(e_i)| < \frac{1}{2^{p+2}}.$$

(xi) From (ii) and (iii) with $n = q$ and $l = \beta(m,q,q)$, we have

$$|f(z_{mq}) - s_{mq\beta(m,q,q)}| < \frac{1}{2^q}$$

or

$$|f(z_{mq}) - t_{mq}| < \frac{1}{2^q}.$$

(xii) Define

$$\delta(m,p) := \max(\nu(m, \varepsilon_0(m,i,p)), p+2),$$

and notice that

$$q \geq \delta(m,p) \to q \geq p+2 \to \frac{1}{2^q} \leq \frac{1}{2^{p+2}},$$

$$q \geq \delta(m,p) \to q \geq \nu(m, \varepsilon_0(m,i,p)).$$

(xiii) Summing up (ix)$\sim$(xii), we obtain, presuming that $q \geq \delta(m,p)$,

$$|f(x_m) - t_{mq}| \leq |f(x_m) - f(e_i)| + |f(e_i) - f(z_{mq})| + |f(z_{mq}) - t_{mq}|$$

$$< \frac{3}{2^{p+2}} < \frac{1}{2^p}.$$

$\{t_{mq}\}$ is a recursive sequence of rational numbers. Since $i = \iota(m,p)$ is recursive in ν, so is $\delta(m,p)$. (xiii) therefore proves that $\{f(x_m)\}$ is weakly E-computable by $\{t_{mq}\}$ and δ.

Note 1) Recall that γ is recursive and ι and ε are recursive in ν. So, δ is obtained by repeated substitutions of ν. It is known, and is explicitly explained in [14] that the family of limiting recursive functions is closed under substitutions. (This does not mean that it is closed with respect to repeated applications of the limit operation.)

2) The conditions in $\mathcal{D}$-2 and $\mathcal{D}$-3 may appear somewhat arbitrary. However, those properties in fact hold for the examples in preceding works (cf. [12],[14]). For example, in [14], $\varepsilon(m,i,n) = n$ for $\mathcal{D}$-3. In [12], the theorem holds without even assuming that f be $\mathcal{U}$-continuous.

6. An example

There are many examples of Euclidean discontinuous functions whose sequential computabilities have been successfully treated; among them are the floor function and the Rademacher functions ([6], [10], [16]) as well as Brattka's Fine continuous function ([1]). Brattka's function is an example of a Fine continuous but not locally uniformly Fine continuous function.

Here we explain how the requirements of our framework are met and how sequential computability can be established with an easy example of the floor function $[x]$, by partly reviving the corresponding content in [10].

Recall that the value $[x]$ is an integer, a computable number, for any real number x. There is thus no sense in questioning about the computability of the function value at a single point x. It *is* computable. With a sequence of values, it takes on a new aspect.

In [10], an E-computable sequence of real numbers $\{b_m\}$ such that the sequence $\{[b_m]\}$ is not E-computable has been constructed. It is defined as follows. Let a be a recursive injection whose range is not recursive.

$$b_m = \begin{cases} 1 - \frac{1}{2^l} & \text{if } m = a(l) \text{ for some (unique) } l, \\ 1 & \text{otherwise.} \end{cases}$$

$\{b_m\}$ is E-computable, since it is effectively approximated by the recursive sequence of rational numbers $\{r_{mk}\}$ defined below.

$$r_{mk} = \begin{cases} 1 - \frac{1}{2^l} & \text{if } m = a(l) \text{ for some } l \leq k, \\ 1 & \text{otherwise.} \end{cases} \tag{3}$$

From the definition we have

$$[b_m] = \begin{cases} 0 & \text{if } m = a(l) \text{ for some } l, \\ 1 & \text{otherwise.} \end{cases}$$

If $\{[b_m]\}$ were E-computable, then the range of a would be recursive, yielding a contradiction. So, $\{[b_m]\}$ cannot be an E-computable sequence.

This counter-example assures us of the following fact: the floor function $[x]$ does not necessarily preserve E-sequential computability.

With the function $[x]$, we associate a uniform space $\mathcal{U} = \langle \mathbf{R}, \{U_n\} \rangle$ by mutually isolating the half-open intervals $[l, l+1)$ for all integer l. Namely,

$$U_n(x) = (x - \frac{1}{2^n}, x + \frac{1}{2^n}) \cap [l, l+1) \quad \text{if} \quad x \in [l, l+1), n = 0, 1, 2, \cdots.$$

It is easy to prove that $\mathcal{U} = \langle \mathbf{R}, \{U_n\} \rangle$ forms an effective uniform space. (This uniformity is different from that of [6] associated with the same function $[x]$.)

The computability structure is defined by taking recursive sequences of rational numbers and by adding the $\mathcal{U}$-effective limits of recursive double sequences of rational numbers. Then it follows immediately that the assumption $[\mathcal{A}]$ is satisfied.

Note The sequence $\{b_m\}$ defined above is not $\mathcal{U}$-computable, for suppose $\{b_m\}$ were $\mathcal{U}$-computable. Then there would be a recursive sequence of rational numbers, say $\{s_{mk}\}$, and a recursive function γ such that $\forall k \geq \gamma(m,n). s_{mk} \in U_n(b_m)$, with which holds

$$s_{m,\gamma(m,0)} \in [0,1) \leftrightarrow b_m < 1 \leftrightarrow m \in \text{ range of } a.$$

Since the left-hand side is decidable (recursive in m), so must be the right-hand side. But then the range of the function a would be recursive, contradicting the property of a.

Proposition 6.1. An E-computable sequence of real numbers is weakly $\mathcal{U}$-computable, and hence the condition $[\mathcal{C}]$ holds.

Proof We can prove the proposition in a manner similar to the corresponding proposition in Section 3 of [14]. Suppose $\{x_m\}$ is E-computable with a recursive sequence of rational numbers $\{r_{mp}\}$ and a recursive modulus of convergence α. It is known that these data induce a recursive non-increasing sequence of rationals which effectively converges to $\{x_m\}$. So we might as well assume that $\{r_{mp}\}$ and α have such a property. We can then define a recursive function $\kappa(m,p)$ such that $r_{mp} \in [k, k+1)$ with $k = \kappa(m,p)$. $\kappa(m,p)$ is non-increasing and eventually constant with respect to p. If we put $k_m = \lim_p \kappa(m,p)$, then $x_m \in [k_m, k_m + 1)$.

Define a recursive function ν_0 as follows.

$$\nu_0(m,1) = 1;$$

$$\nu_0(m,p+1) = \nu_0(m,p) \quad \text{if} \quad \kappa(m,p+1) = \kappa(m,p);$$

$$\nu_0(m,p+1) = p+1 \quad \text{if} \quad \kappa(m,p+1) < \kappa(m,p);$$

Then $\nu_0(m,p)$ is non-decreasing and becomes eventually constant. So, $\nu(m) = \lim_p \nu_0(m,p)$ is a limiting recursive function. Define

$$\beta(m,n) = \max(\alpha(m,n), \nu(m)).$$

β is recursive in ν, hence is limiting recursive, and satisfies that $|x_m - r_{mp}| < \frac{1}{2^n}$ and $r_{mp} \in [k_m, k_m + 1)$ if $p \geq \beta(m,n)$, or $r_{mp} \in U_n(x_m)$, and hence $\{x_m\}$ is weakly E-computable by $\{r_{mp}\}$ and $\beta(m,n)$.

Note One must be careful in reading this proof. Although we have the formula $x_m \in [k_m, k_m + 1)$, we need *not* "compute" $\{k_m\}$. $\{k_m\}$ is not involved in the construction of β. Each k_m is used only in claiming that x_m and r_{mp} belong to the same half interval without referring to any computation.

Corollary 1. For any E-computable sequence $\{x_m\}$, there is a sequence of integers $\{j_m\}$ which is recursive in ν (defined in the proof above) such that $x_m \in [j_m, j_m + 1)$.

Proof For the sequence of rational numbers $\{r_{mp}\}$ above, one can effectively determine $\{q_{mp}\}$ so that $r_{mp} \in [q_{mp}, q_{mp} + 1)$. Since $r_{mp} \in U_n(x_m)$ if $p \geq \beta(m,n)$, we can take $j_m = q_{m,\beta(m,0)}$.

Note The fact that the sequence $\{b_m\}$ defined above is weakly $\mathcal{U}$-computable can be shown as follows (whose treatment is slightly different from the general construction in the proposition). As for $\{z_{mk}\}$, it suffices to take $\{r_{mk}\}$ in (3). Since this is a recursive sequence of rational numbers, it is $\mathcal{U}$-computable by $\mathcal{A}$-1. Define a function h as follows.

$$h(m,k) = 1 \quad \text{if} \quad \forall l \leq k . r_{ml} = 1;$$

$$h(m,k) = k_0 + 1 \quad \text{if } k_0 \text{ is the least } l \leq k . r_{ml} < 1.$$

h is recursive, and it is easy to see that $\nu(m) = \lim_k h(m,k)$ exists. $\nu(m) = 1$ or $= k_0 + 1$, and ν serves as a limiting recursive modulus of convergence of $\{r_{mn}\}$ to $\{b_m\}$.

Proposition 6.2. (Condition $\mathcal{D}$ for $[x]$) The floor function $[x]$ satisfies the condition $\mathcal{D}$.

Proof $\mathcal{D}$-1: For the sequence $\{e_i\}$, we can take $\{l\}$, the set of all integers. We will take l itself as an index, instead of the i of e_i in the general case (by extending relevant functions accordingly). Put $\gamma(l,p) = 0$ (a constant function). Then $x \in U_{\gamma(l,p)}(l) = U_0(l) = [l, l+1)$ implies $[x] - [l] = 0$.

$\mathcal{D}$-2: Let $\{x_m\}$ be an E-computable sequence. We can take j_m in Corollary after Proposition 6.1 as $\iota(m,p)$ for all p.

$\mathcal{D}$-3: Let $\{x_m\}$ and $\{j_m\}$ be as above. Evaluate a $q = q_{m,n}$ such that $x_m + \frac{1}{2^q} < j_m + \frac{1}{2^n}$. Now put $\varepsilon(m,i,n) = q_{mn}$ if $i = j_m$; $= 1$ otherwise. ε is recursive in ν, and it is obvious that it satisfies the condition.

Proposition 6.3. ($\mathcal{U}$-sequential computability of $[x]$) The function $[x]$ is $\mathcal{U}$- sequentially computable.

Proof Suppose $\{x_m\}$ is $\mathcal{U}$-computable. Then there are recursive $\{r_{mn}\}$ and α which represent $\{x_m\}$. For $k \geq \alpha(m,0)$, $r_{mk} \in U_0(x_m)$, that is, if $x_m \in [l, l+1)$, then $r_{m\alpha(m,0)} \in [l, l+1)$ and hence $[x_m] = [r_{m\alpha(m,0)}] = l$. Since $\{r_{m\alpha(m,0)}\}$ is a recursive sequence of rational numbers, $\{[r_{m\alpha(m,0)}]\}$ is E-computable (a recursive sequence of integers), hence so is $\{[x_m]\}$.

From the fact that $[\mathcal{A}]$ is satisfied, Propositions 6.1, 6.2 and 6.3 and Theorem 2, we obtain: $[x]$ is $\mathcal{L}$-sequentially computable.

Note 1) With this topology, Proposition 6.2, that is, the condition $[\mathcal{D}]$, is in fact not necessary for the $\mathcal{L}$- sequential computability of $[x]$. It can be proved without the condition $[\mathcal{D}]$ due to the peculiarity of the uniformity $\mathcal{U}$. The situation is similar to Theorem 3 of [12]. We have proved Proposition 6.2 in order to show how the condition $[\mathcal{D}]$ holds for this uniformity.

2) $\mathcal{L}$-sequential computability of $[x]$ has been directly shown in detail in Section 2 of [10].

Now, an attempt of computing $\{[b_m]\}$ goes very roughly as follows (cf. [10],[8]). It holds that $0 < b_m < 2$ for all m. In order to determine whether $0 < b_m < 1$ or $b_m \geq 1$, we define a recursive sequence of rational numbers (integers as a matter of fact) $\{N_{mp}\}$ as follows.

$$N_{mp} = \begin{cases} 1 & \text{if } r_{m\alpha(m,p)} \geq 1 - \frac{1}{2^p}, \\ 0 & \text{if } r_{m\alpha(m,p)} < 1 - \frac{1}{2^p}. \end{cases} \tag{4}$$

So, $\{[b_m]\}$ is weakly computable by $\{N_{mp}\}$ with a modulus of convergence which is recursive in ν (cf. Proposition 6.1).

Remark We can define a metric d on $\mathbf{R}$ by putting

$$d(x,y) = |x-y| \quad \text{if} \quad x, y \in [l, l+1) \quad \text{for some} \quad l;$$

$$d(x,y) = 1 \quad \text{otherwise.}$$

It can be shown that $\langle \mathbf{R}, d \rangle$ can be endowed with a computability structure and d is equivalent with $\mathcal{U}$ with respect to effective convergence (cf. [5] and [13]). The floor function is a computable function in this metric space.

7. Limiting recursion versus effective uniformity

The notion of $\mathcal{L}$-sequential computability and that of $\mathcal{U}$-sequential computability appear mutually quite different. Let us see this with the example $[x]$. As has been mentioned in 2) of Note after Proposition 6.3, $\mathcal{L}$-sequential computability of $[x]$ has been analyzed in [10] in detail. Given an E-computable sequence $\{x_m\}$, one attempts to compute the values $\{[x_m]\}$ step by step, but one can obtain the value only after a complete process of evaluation ad infinitum and with an observation of the entire process. The "computation from the infinity" corresponds to accepting the limit of a recursive process.

On the other hand, with the uniform topology, one does not even attempt to compute the function value for certain E-computable sequences such as $\{b_m\}$ in Section 6. For each l, only the interval $[l, l+1)$ is in one's sight, and one knows that there $[x] = l$.

Considering these situations, the mathematical equivalence of two notions of sequential computability of a function seems to need some speculations.

Let us first observe the limiting recursion method. Here one attempts to compute the function values mechanically. The input values for a function are supplied with a recursive sequence of rationals and a recursive modulus of convergence, but the outputs, viz. the function values, are represented by a recursive sequence of rationals with a limiting recursive modulus of convergence, and the result cannot be improved. This has been discussed in [10] and [8] in detail.

The merit of this method lies in its simplicity. The only tool in need beyond the recursive function is taking the limit of a recursive function. The function value at a point of discontinuity is represented with a recursive sequence of rational numbers. At each step of computation one is approaching the right value, and one knows that eventually one gets the proper value, though not knowing how close we are to the value. This may be a bit tantalizing if one wishes to know where one is now. It is, however, assuring and in a way sufficient to know that one stands on the right track. It is along the straight extension of the computation of continuous functions; only the speed of convergence is modulated by limiting recursion instead of recursion. The idea is simple and easily understood. No extra knowledge is required. This is its advantage. A disadvantage may be that it does not seem to represent the mental activity of a mathematician computing the function value at a point of discontinuity.

In the method of effective uniformity, with each function a uniform space in which it becomes continuous is associated. The theory of computability structure in such a space can be developed in a general setting, and a function is defined to be computable as a continuous function in this topology. We can thus adhere to the computability problem of a continuous function. Except for recursive functions, we do not need any special tool beyond ordinary mathematical knowledge. For each instance of a function, we only need to associate a uniformity by isolating the points of discontinuity or the intervals determined by the points of discontinuity. This approach is also quite intuitive. The values of a function in each interval (possibly consisting of a singleton) of continuity can be computed as in the case of a Euclidean-continuous function.

In the effective uniformity method, it is important to notice that one can recognize the points of discontinuity "intuitively." In the case of $[x]$, these are integers, and they are the most obvious points that a human being can recognize on the real line. Consider another example. Let τ denote the function which coincides with the tan function where tan is defined, and takes the value 0 where tan is not defined. The computation of τ at a point of discontinuity like $\frac{\pi}{2}$, which should cause a problem in a mechanical computation, is the easier part: the value is 0. Isolating the such points (and the computation of the function values at them), though not a decidable procedure, is thus intuitively appealing. The effective uniformity method thus describes the human mental activity of computing a Euclidean-discontinuous function.

We need not attempt to judge effectively if a real number is a point of discontinuity. The judgement is taken care of in the definition of the effective uniformity; $U_n(\frac{\pi}{2}) = \{\frac{\pi}{2}\}$ for τ, for example. We have also seen another example $[x]$. It is a mathematical activity, and we are at liberty to do that. In that sense, the theory of effective uniformity yields a "supple method" (according to the phrasing of Nakatogawa) for computing a discontinuous function. It reflects the flexibility that a mathematician wishes to practice and experience.

It is Theorems 1 and 2 that related the two methods.

Acknowledgment The first author is indebted to K. Nakatogawa for many hours of discussion on the notion of computability and to S. Hayashi for having introduced Gold's theory to her. Our gratitude goes to T. Mori and Y. Tsujii for many occasions of mathematical discussion.

References

[1] Brattka, V., *Some Notes on Fine Computability*, JUCS, 8(2002), 382-395.

[2] Gold, E.M., *Limiting recursion*, JSL, 30-1 (1965), 28-48.

[3] Kawada, Y., Mimura, Y., *Outline of Modern Mathematics* (in Japanese), Iwanami Shoten (1965).

[4] Pour-El, M.B., Richards, J.I., *Computability in Analysis and Physics*, Springer-Verlag (1989): available also at Projecteuclid.

[5] Mori, T., Tsujii. Y. and Yasugi, M., *Computability structures on metric spaces*, Combinatorics, Complexity, and Logic, Springer(1996), 351-362.

[6] Tsujii, Y., Yasugi, M., Mori, T., *Some properties of the effectively uniform topological space*, Computability and Complexity in Analysis, LNCS 2064, Springer (2001) 336-356.

[7] Tsujii, Y., Yasugi, M., Mori, T., *Sequential computability of a function - Diagonal space and limiting recursion -* , ENTCS 120(2005),187-199.

[8] Yasugi, M., *Limiting computability of discontinuous functions-significance and problems*, J. of Japan Assoc. for Phil. of Sci., vol.30,no2(2003), 13-18 (in Japanese).

[9] Yasugi, M., *On notions of computation over the continuum - Beyond recursive functions -* , The Kyoto Graduate Journal for Philosophy (Tetsugakuronso) XXXV(2008), The Kyoto Graduate Society for Philosophy, Kyoto University, 199-209 (in Japanese).

[10] Yasugi, M., Brattka, V., Washihara, M., *Computability aspects of some discontinuous functions*, Scientiae Mathematicae Japonicae (SCMJ) Online, vol.5 (2001), 405-419.

[11] Yasugi, M. Mori, T., Tsujii, Y., *The effective sequence of uniformities and its limit: as a methodology in computable analysis*, Annals of Philosophy of Science, vol.15, no.2 (2007), 99-121(47-69).

[12] Yasugi, M., Tsujii, Y., *Computability of a function with jumps-Effective uniformity and limiting recursion-*, Topology and its Applications (Elsevier Science), 146-147(2005), 563-582.

[13] M.Yasugi, Y.Tsujii and T.Mori, *Metrization of the uniform space and effective convergence*, Math. Log. Quart. 48(2002) Suppl. 1, 123-130.

[14] Yasugi, M., Tsujii, Y., Mori, T., *Sequential computability of a function - Effective Fine space and limiting recursion -*, JUCS 11-12(2005), 2179-2191;
http://www.jucs.org/

[15] Yasugi, M., Washihara, M., *Computability structures in analysis*, Sugaku Expositions (AMS) 13 (2000) no.2, 215-235.

[16] Yasugi, Y., Washihara, M., *A note on Rademacher functions and computability*, Words, Languages and Combinatorics III, World Scientific (2003) 466-475.

*Kyoto Sangyo University and Graduate School of Kyoto University, Kamigamomotoyama, Kita-Ku, Kyoto 603-8555 Japan
E-mail address: yasugi@cc.kyoto-su.ac.jp

**Kyoto Sangyo University, Kamigamomotoyama, Kita-Ku, Kyoto 603-8555 Japan
E-mail address: RXK00312@nifty.com