Subrecursive degrees of difference representations of irrational numbers

Ivan Georgiev*

Sofia University "St. Kliment Ohridski"

Computability and Complexity in Analysis

Kyoto, Japan, 24-26 September 2025

*This work is supported by NextGenerationEU, through the NRRP of Bulgaria, project no. BG-RRP-2.004-0008-C01.

Overview of the talk

We consider different ways to represent irrational numbers:

Overview of the talk

We consider different ways to represent irrational numbers:

- base expansions

Overview of the talk

We consider different ways to represent irrational numbers:

- base expansions
- Dedekind cuts

Overview of the talk

We consider different ways to represent irrational numbers:

- base expansions
- Dedekind cuts
- ...

Overview of the talk

We consider different ways to represent irrational numbers:

- base expansions
- Dedekind cuts
- ...

They are all computably equivalent: any two of them can be computably transformed into one another, uniformly in the represented number.

Overview of the talk

We consider different ways to represent irrational numbers:

- base expansions
- Dedekind cuts
- ...

They are all computably equivalent: any two of them can be computably transformed into one another, uniformly in the represented number.

Our main question of interest: *can we transform one representation into another without using unbounded search*?

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

We will denote $R_1 \preccurlyeq_S R_2$ (R_1 *is subrecursive in* R_2) if there exists an algorithm A with an oracle, such that:

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

We will denote $R_1 \preccurlyeq_S R_2$ (R_1 *is subrecursive in* R_2) if there exists an algorithm A with an oracle, such that:

- for any ϕ, which is an R_2-representation of an irrational $\alpha \in (0,1)$, A^{ϕ} is an R_1-representation of the same α;

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

We will denote $R_1 \preccurlyeq_S R_2$ (R_1 *is subrecursive in* R_2) if there exists an algorithm A with an oracle, such that:

- for any ϕ, which is an R_2-representation of an irrational $\alpha \in (0,1)$, A^ϕ is an R_1-representation of the same α;
- A uses no unbounded search.

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

We will denote $R_1 \preccurlyeq_S R_2$ (R_1 *is subrecursive in* R_2) if there exists an algorithm A with an oracle, such that:

- for any ϕ, which is an R_2-representation of an irrational $\alpha \in (0, 1)$, A^ϕ is an R_1-representation of the same α;
- A uses no unbounded search.

We will also denote

$$R_1 \equiv_S R_2 \;\text{ if }\; R_1 \preccurlyeq_S R_2 \;\&\; R_2 \preccurlyeq_S R_1$$

$$R_1 \prec_S R_2 \;\text{ if }\; R_1 \preccurlyeq_S R_2 \;\&\; R_2 \not\preccurlyeq_S R_1.$$

Subrecursive reducibility

Let R_1 and R_2 be representations of irrational numbers.

We will denote $R_1 \preccurlyeq_S R_2$ (R_1 *is subrecursive in* R_2) if there exists an algorithm A with an oracle, such that:

- for any ϕ, which is an R_2-representation of an irrational $\alpha \in (0,1)$, A^ϕ is an R_1-representation of the same α;
- A uses no unbounded search.

We will also denote

$$R_1 \equiv_S R_2 \;\text{ if }\; R_1 \preccurlyeq_S R_2 \;\&\; R_2 \preccurlyeq_S R_1$$

$$R_1 \prec_S R_2 \;\text{ if }\; R_1 \preccurlyeq_S R_2 \;\&\; R_2 \not\preccurlyeq_S R_1.$$

The equivalence classes of representations are called S-degrees.

Examples for Representations

For an irrational number $\alpha \in (0, 1)$:

Examples for Representations

For an irrational number $\alpha \in (0, 1)$:

- the *Dedekind cut* of α is the function $D : \mathbb{Q} \to \{0, 1\}$, such that

$$D(q) = \begin{cases} 0, & \text{if } q < \alpha, \\ 1, & \text{if } q > \alpha. \end{cases}$$

Examples for Representations

For an irrational number $\alpha \in (0,1)$:

- the *Dedekind cut* of α is the function $D : \mathbb{Q} \to \{0,1\}$, such that

$$D(q) = \begin{cases} 0, & \text{if } q < \alpha, \\ 1, & \text{if } q > \alpha. \end{cases}$$

- for $b \geq 2$, the *base-b expansion* of α is the function $E_b : \mathbb{N} \to \{0,1,\ldots,b-1\}$, such that

$$\alpha \;=\; \sum_{n=0}^{\infty} E_b(n) \cdot b^{-n}.$$

$E_b \preccurlyeq_S D$

Assume we have computed $E_b(1), E_b(2), \ldots, E_b(n)$ and let

$$q_n \ = \ E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

$E_b \preccurlyeq_S D$

Assume we have computed $E_b(1), E_b(2), \ldots, E_b(n)$ and let

$$q_n = E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

To compute $E_b(n+1)$: we search for the unique $\mathtt{D} \in \{0, 1, \ldots, b-1\}$, such that

$$D(q_n + \mathtt{D} \cdot b^{-n-1}) = 0 \quad \& \quad D(q_n + (\mathtt{D}+1) \cdot b^{-n-1}) = 1$$

and then $E_b(n+1) = \mathtt{D}$.

$E_b \preccurlyeq_S D$

Assume we have computed $E_b(1), E_b(2), \ldots, E_b(n)$ and let

$$q_n \; = \; E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

To compute $E_b(n+1)$: we search for the unique $\mathtt{D} \in \{0, 1, \ldots, b-1\}$, such that

$$D(q_n + \mathtt{D} \cdot b^{-n-1}) = 0 \quad \& \quad D(q_n + (\mathtt{D}+1) \cdot b^{-n-1}) = 1$$

and then $E_b(n+1) = \mathtt{D}$.

No unbounded search is used in this algorithm!

$D \not\leq_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

$D \not\leq_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

If q has a finite base-b expansion of length n, then

$$q \; < \; \alpha \iff q \; \leq \; E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

$D \not\leq_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

If q has a finite base-b expansion of length n, then

$$q \ < \ \alpha \iff q \ \leq \ E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

But what if q has an infinite base-b expansion? For example, let $b = 10$ and $q = 1/3 = 0.3333333\ldots$.

$D \not\leq_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

If q has a finite base-b expansion of length n, then

$$q \ < \ \alpha \iff q \ \leq \ E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

But what if q has an infinite base-b expansion? For example, let $b = 10$ and $q = 1/3 = 0.3333333\ldots$. To decide whether $q \ < \ \alpha$ we must search for a position n, such that $E_{10}(n) \neq 3$.

$D \not\leq_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

If q has a finite base-b expansion of length n, then

$$q < \alpha \iff q \leq E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

But what if q has an infinite base-b expansion? For example, let $b = 10$ and $q = 1/3 = 0.3333333\ldots$. To decide whether $q < \alpha$ we must search for a position n, such that $E_{10}(n) \neq 3$.

This algorithm requires unbounded search!

$D \not\prec_S E_b$

Given $q \in \mathbb{Q}$, we want to decide whether $q < \alpha$ by using access to the base-b expansion E_b of α.

If q has a finite base-b expansion of length n, then

$$q \ < \ \alpha \iff q \ \leq \ E_b(1) \cdot b^{-1} + E_b(2) \cdot b^{-2} + \ldots + E_b(n) \cdot b^{-n}.$$

But what if q has an infinite base-b expansion? For example, let $b = 10$ and $q = 1/3 = 0.3333333\ldots$. To decide whether $q \ < \ \alpha$ we must search for a position n, such that $E_{10}(n) \neq 3$.

This algorithm requires unbounded search!

Therefore, we have $E_b \prec_S D$.

How to formally prove $R_1 \not\precsim_S R_2$?

There must exist a time bound t,

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s,

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s, one can construct an irrational $\alpha \in (0, 1)$ with the following properties:

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s, one can construct an irrational $\alpha \in (0, 1)$ with the following properties:

- α has at least one R_2-representation computable in time $O(t)$;
- no R_1-representation of α is computable in time $O(s)$.

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s, one can construct an irrational $\alpha \in (0, 1)$ with the following properties:

- α has at least one R_2-representation computable in time $O(t)$;
- no R_1-representation of α is computable in time $O(s)$.

In many cases, t can be chosen primitive recursive.

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s, one can construct an irrational $\alpha \in (0,1)$ with the following properties:

- α has at least one R_2-representation computable in time $O(t)$;
- no R_1-representation of α is computable in time $O(s)$.

In many cases, t can be chosen primitive recursive.

Sometimes α is constructed by diagonalization against all R_1-representations computable in time $O(s)$.

How to formally prove $R_1 \not\preceq_S R_2$?

There must exist a time bound t, such that for any time bound s, one can construct an irrational $\alpha \in (0,1)$ with the following properties:

- α has at least one R_2-representation computable in time $O(t)$;
- no R_1-representation of α is computable in time $O(s)$.

In many cases, t can be chosen primitive recursive.

Sometimes α is constructed by diagonalization against all R_1-representations computable in time $O(s)$.

In other cases, as we shall see, α has a more transparent definition, justified using a growth argument.

Base-b sum approximations from below

Let us fix some base b. Any irrational number $\alpha \in (0,1)$ can be written in the form

$$\alpha = 0 + \frac{d_1}{b^{k_1}} + \frac{d_2}{b^{k_2}} + \frac{d_3}{b^{k_3}} + \dots,$$

where k_n is a strictly increasing sequence of positive integers and d_n are non-zero base-b digits, $d_n \in \{1, \dots, b-1\}$.

Base-b sum approximations from below

Let us fix some base b. Any irrational number $\alpha \in (0, 1)$ can be written in the form

$$\alpha = 0 + \frac{d_1}{b^{k_1}} + \frac{d_2}{b^{k_2}} + \frac{d_3}{b^{k_3}} + \dots,$$

where k_n is a strictly increasing sequence of positive integers and d_n are non-zero base-b digits, $d_n \in \{1, \dots, b - 1\}$.

The function $\hat{A}_b$, defined by $\hat{A}_b(n) = d_n b^{-k_n}$ for $n > 0$ and $\hat{A}_b(0) = 0$ is called the *base-b sum approximation from below* of the number α.

Base-b sum approximations from above

Moreover, we can write

$$\alpha = 1 - \frac{d_1'}{b^{m_1}} - \frac{d_2'}{b^{m_2}} - \frac{d_3'}{b^{m_3}} - \dots,$$

where m_n is a strictly increasing sequence of positive integers and d_n' are non-zero base-b digits.

Base-b sum approximations from above

Moreover, we can write

$$\alpha = 1 - \frac{d'_1}{b^{m_1}} - \frac{d'_2}{b^{m_2}} - \frac{d'_3}{b^{m_3}} - \dots,$$

where m_n is a strictly increasing sequence of positive integers and d'_n are non-zero base-b digits.

The function $\check{A}_b$, defined by $\check{A}_b(n) = d'_n b^{-m_n}$ for $n > 0$ and $\check{A}_b(0) = 1$ is called the *base-b sum approximation from above* of the number α.

Example in base 10

For example, let us have $\alpha = 0.07319990004\ldots$ in base $b = 10$.

Example in base 10

For example, let us have $\alpha = 0.07319990004\ldots$ in base $b = 10$. Then

$$\alpha = \frac{7}{10^2} + \frac{3}{10^3} + \frac{1}{10^4} + \frac{9}{10^5} + \frac{9}{10^6} + \frac{9}{10^7} + \frac{4}{10^{11}} + \ldots,$$

Example in base 10

For example, let us have $\alpha = 0.07319990004\ldots$ in base $b = 10$. Then

$$\alpha = \frac{7}{10^2} + \frac{3}{10^3} + \frac{1}{10^4} + \frac{9}{10^5} + \frac{9}{10^6} + \frac{9}{10^7} + \frac{4}{10^{11}} + \ldots,$$

thus

$$\hat{A}_{10}(1) = \frac{7}{10^2}, \ \ \hat{A}_{10}(2) = \frac{3}{10^3}, \ \ \ldots$$

Example in base 10

For example, let us have $\alpha = 0.07319990004\ldots$ in base $b = 10$. Then

$$\alpha = \frac{7}{10^2} + \frac{3}{10^3} + \frac{1}{10^4} + \frac{9}{10^5} + \frac{9}{10^6} + \frac{9}{10^7} + \frac{4}{10^{11}} + \ldots,$$

thus

$$\hat{A}_{10}(1) = \frac{7}{10^2}, \quad \hat{A}_{10}(2) = \frac{3}{10^3}, \quad \ldots$$

Moreover,

$$\alpha = 1 - \frac{9}{10^1} - \frac{2}{10^2} - \frac{6}{10^3} - \frac{8}{10^4} - \frac{9}{10^8} - \frac{9}{10^9} - \frac{9}{10^{10}} - \frac{5}{10^{11}} - \ldots,$$

Example in base 10

For example, let us have $\alpha = 0.07319990004\ldots$ in base $b = 10$. Then

$$\alpha = \frac{7}{10^2} + \frac{3}{10^3} + \frac{1}{10^4} + \frac{9}{10^5} + \frac{9}{10^6} + \frac{9}{10^7} + \frac{4}{10^{11}} + \ldots,$$

thus

$$\hat{A}_{10}(1) = \frac{7}{10^2}, \quad \hat{A}_{10}(2) = \frac{3}{10^3}, \quad \ldots$$

Moreover,

$$\alpha = 1 - \frac{9}{10^1} - \frac{2}{10^2} - \frac{6}{10^3} - \frac{8}{10^4} - \frac{9}{10^8} - \frac{9}{10^9} - \frac{9}{10^{10}} - \frac{5}{10^{11}} - \ldots,$$

thus

$$\check{A}_{10}(1) = \frac{9}{10^1}, \quad \check{A}_{10}(2) = \frac{2}{10^2}, \quad \ldots$$

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$.

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$. Proof?

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$. Proof?

For a time bound s, let $d : \mathbb{N} \to \mathbb{N}$ be a strictly increasing function with primitive recursive graph, such that d is not computable in time $O(s)$.

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$. Proof?

For a time bound s, let $d : \mathbb{N} \to \mathbb{N}$ be a strictly increasing function with primitive recursive graph, such that d is not computable in time $O(s)$.

Consider the following irrational number:

$$\alpha = 0.000 \ldots .0001000 \ldots .0001000 \ldots .0001000 \ldots ..$$

$$\uparrow d_0 \qquad\qquad \uparrow d_1 \qquad\qquad \uparrow d_2$$

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$. Proof?

For a time bound s, let $d : \mathbb{N} \to \mathbb{N}$ be a strictly increasing function with primitive recursive graph, such that d is not computable in time $O(s)$.

Consider the following irrational number:

$$\alpha = 0.000\ldots.0001000\ldots.0001000\ldots.0001000\ldots..$$
$$\uparrow d_0 \qquad\qquad \uparrow d_1 \qquad\qquad \uparrow d_2$$

Then we have: E_b is primitive recursive, $\hat{A}_b$ is not computable in time $O(s)$ and $\check{A}_b$ is primitive recursive.

Results on sum approximations

It is evident that $E_b \preccurlyeq_S \hat{A}_b$ and $E_b \preccurlyeq_S \check{A}_b$.

But in fact we have $E_b \prec_S \hat{A}_b$ and $E_b \prec_S \check{A}_b$. Proof?

For a time bound s, let $d : \mathbb{N} \to \mathbb{N}$ be a strictly increasing function with primitive recursive graph, such that d is not computable in time $O(s)$.

Consider the following irrational number:

$$\alpha = 0.000\ldots.0001000\ldots.0001000\ldots.0001000\ldots..$$
$$\uparrow d_0 \qquad\qquad \uparrow d_1 \qquad\qquad \uparrow d_2$$

Then we have: E_b is primitive recursive, $\hat{A}_b$ is not computable in time $O(s)$ and $\check{A}_b$ is primitive recursive.

For any base b, the representations $\hat{A}_b, \check{A}_b$ and D are pairwise incomparable with respect to $\preccurlyeq_S$.

General sum approximations

Let $\alpha \in (0, 1)$ be an irrational number.

General sum approximations

Let $\alpha \in (0, 1)$ be an irrational number.

The function $\hat{G} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, such that

$$\hat{G}(b, n) = \hat{A}_b(n), \quad \hat{G}(b, n) = 0 \text{ for } b < 2$$

will be called *general sum approximation from below* of α.

General sum approximations

Let $\alpha \in (0,1)$ be an irrational number.

The function $\hat{G} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, such that

$$\hat{G}(b,n) = \hat{A}_b(n), \quad \hat{G}(b,n) = 0 \text{ for } b < 2$$

will be called *general sum approximation from below* of α.

The function $\check{G} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, such that

$$\check{G}(b,n) = \check{A}_b(n), \quad \check{G}(b,n) = 0 \text{ for } b < 2$$

will be called *general sum approximation from above* of α.

General sum approximations

Let $\alpha \in (0,1)$ be an irrational number.

The function $\hat{G} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, such that

$$\hat{G}(b,n) = \hat{A}_b(n), \quad \hat{G}(b,n) = 0 \text{ for } b < 2$$

will be called *general sum approximation from below* of α.

The function $\check{G} : \mathbb{N} \times \mathbb{N} \to \mathbb{Q}$, such that

$$\check{G}(b,n) = \check{A}_b(n), \quad \check{G}(b,n) = 0 \text{ for } b < 2$$

will be called *general sum approximation from above* of α.

In [5] we constructed, given an arbitrary time bound s,
an irrational α such that $\hat{A}_b$ is primitive recursive for any fixed b,
but $\hat{G}$ is not computable in time $O(s)$.

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \;\; \Rightarrow \;\; q < T^{\uparrow}(q) < \alpha;$$

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \;\; \Rightarrow \;\; q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \;\; \Rightarrow \;\; q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \ \Rightarrow \ q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:
1. $q < \alpha \ \Leftrightarrow \ q \leq \hat{G}(2n, 1)$;

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \quad \Rightarrow \quad q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:
1. $q < \alpha \;\Leftrightarrow\; q \leq \hat{G}(2n, 1)$;
2. $T^{\uparrow}(q) = \hat{G}(2n, 1) + \hat{G}(2n, 2)$.

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \quad \Rightarrow \quad q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:
1. $q < \alpha \;\Leftrightarrow\; q \leq \hat{G}(2n, 1)$;
2. $T^{\uparrow}(q) = \hat{G}(2n, 1) + \hat{G}(2n, 2)$.

Conversely, if we have D and some $T^{\uparrow}$, we can compute $\hat{G}(b, n)$ using primitive recursion on n.

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \quad \Rightarrow \quad q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:
1. $q < \alpha \;\Leftrightarrow\; q \leq \hat{G}(2n, 1)$;
2. $T^{\uparrow}(q) = \hat{G}(2n, 1) + \hat{G}(2n, 2)$.

Conversely, if we have D and some $T^{\uparrow}$, we can compute $\hat{G}(b, n)$ using primitive recursion on n.

A trace function from above for α is a function $T^{\downarrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q > \alpha \quad \Rightarrow \quad q > T^{\downarrow}(q) > \alpha.$$

Trace functions

A trace function from below for α is a function $T^{\uparrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q < \alpha \;\; \Rightarrow \;\; q < T^{\uparrow}(q) < \alpha;$$

Given $\hat{G}$ we can produce the Dedekind cut D and $T^{\uparrow}$.
For $q \in \mathbb{Q} \cap [0,1]$ with denominator n:
1. $q < \alpha \;\Leftrightarrow\; q \leq \hat{G}(2n, 1)$;
2. $T^{\uparrow}(q) = \hat{G}(2n, 1) + \hat{G}(2n, 2)$.

Conversely, if we have D and some $T^{\uparrow}$, we can compute $\hat{G}(b, n)$ using primitive recursion on n.

A trace function from above for α is a function $T^{\downarrow} : \mathbb{Q} \cap [0,1] \to \mathbb{Q} \cap [0,1]$, such that:

$$q > \alpha \;\; \Rightarrow \;\; q > T^{\downarrow}(q) > \alpha.$$

By symmetry, $\check{G}$ is S-equivalent to D and $T^{\downarrow}$.

Difference representations with respect to rational numbers

Let R be a representation of *all real numbers*, such that any $\alpha \in [0, 1]$ has a unique representation $R^{\alpha} : A \to B$.

Difference representations with respect to rational numbers

Let R be a representation of *all real numbers*, such that any $\alpha \in [0,1]$ has a unique representation $R^{\alpha} : A \to B$.

For any $\alpha \in (0,1) \setminus \mathbb{Q}$ we define $\mathcal{D}\mathrm{iff}_R^{\alpha} : \mathbb{Q} \to A$ with

$$\mathcal{D}\mathrm{iff}_R^{\alpha}(q) \;=\; \mu a \in A[R^{\alpha}(a) \neq R^{q}(a)].$$

Difference representations with respect to rational numbers

Let R be a representation of *all real numbers*, such that any $\alpha \in [0,1]$ has a unique representation $R^\alpha : A \to B$.

For any $\alpha \in (0,1) \setminus \mathbb{Q}$ we define $\mathcal{D}\mathrm{iff}_R^\alpha : \mathbb{Q} \to A$ with

$$\mathcal{D}\mathrm{iff}_R^\alpha(q) \;=\; \mu a \in A[R^\alpha(a) \neq R^q(a)].$$

Is $\mathcal{D}\mathrm{iff}_R^\alpha$ a representation of irrational numbers?

Two natural assumptions for R

- The restriction of R to rational numbers, $R : \mathbb{Q} \times A \to B$, must be computable.

Two natural assumptions for R

- The restriction of R to rational numbers, $R : \mathbb{Q} \times A \to B$, must be computable.
  This entails that $\mathcal{D}\mathrm{iff}_R^\alpha$ can be computed with oracle R^α.

Two natural assumptions for R

- The restriction of R to rational numbers, $R : \mathbb{Q} \times A \to B$, must be computable.
  This entails that $\mathcal{D}\mathrm{iff}_R^\alpha$ can be computed with oracle R^α.

- There exists a computable function P, such that for any finite list $\overline{a}$ of elements of A and any finite list $\overline{b}$ of elements of B with the same length, $P(\overline{a}, \overline{b}) = q$, where $R^q(a_i) = b_i$ for all i.

Two natural assumptions for R

- The restriction of R to rational numbers, $R : \mathbb{Q} \times A \to B$, must be computable.
  This entails that $\mathcal{D}\mathrm{iff}_R^\alpha$ can be computed with oracle R^α.

- There exists a computable function P, such that for any finite list $\overline{a}$ of elements of A and any finite list $\overline{b}$ of elements of B with the same length, $P(\overline{a}, \overline{b}) = q$, where $R^q(a_i) = b_i$ for all i.

  We can compute R^α with oracle $\mathcal{D}\mathrm{iff}_R^\alpha$ in the following way:
  1. Let $a_0, \ldots, a_k = a$ be the first $k+1$ elements of A and assume we have computed $R^\alpha(a_i)$ for $i < k$.
  2. For each $b_k \in B$:
     Compute $q = P(\overline{a}, \overline{b})$, where $b_i = R^\alpha(a_i)$ for $i < k$.
     If $\mathcal{D}\mathrm{iff}_R^\alpha(q) \neq a$, then return output $R^\alpha(a) = b_k$.

Two natural assumptions for R

- The restriction of R to rational numbers, $R : \mathbb{Q} \times A \to B$, must be computable.
  This entails that $\mathcal{D}\mathrm{iff}_R^\alpha$ can be computed with oracle R^α.

- There exists a computable function P, such that for any finite list $\bar{a}$ of elements of A and any finite list $\bar{b}$ of elements of B with the same length, $P(\bar{a}, \bar{b}) = q$, where $R^q(a_i) = b_i$ for all i.

  We can compute R^α with oracle $\mathcal{D}\mathrm{iff}_R^\alpha$ in the following way:
  1. Let $a_0, \ldots, a_k = a$ be the first $k+1$ elements of A and assume we have computed $R^\alpha(a_i)$ for $i < k$.
  2. For each $b_k \in B$:
     Compute $q = P(\bar{a}, \bar{b})$, where $b_i = R^\alpha(a_i)$ for $i < k$.
     If $\mathcal{D}\mathrm{iff}_R^\alpha(q) \neq a$, then return output $R^\alpha(a) = b_k$.

  If B is finite and P is primitive recursive, $R \preccurlyeq_S \mathcal{D}\mathrm{iff}_R$.

Difference with respect to base-b expansions

As a first example we consider $\mathcal{D}\mathrm{iff}_b$, which corresponds to the base-b expansion E_b. The base-b expansions of numbers of the form $\frac{m}{b^n}$ are assumed to end in zeros.

Difference with respect to base-b expansions

As a first example we consider $\mathcal{D}\mathrm{iff}_b$, which corresponds to the base-b expansion E_b. The base-b expansions of numbers of the form $\frac{m}{b^n}$ are assumed to end in zeros.

Since the range of E_b is finite, we have $E_b \preccurlyeq_S \mathcal{D}\mathrm{iff}_b$.

Difference with respect to base-b expansions

As a first example we consider $\mathcal{D}\text{iff}_b$, which corresponds to the base-b expansion E_b. The base-b expansions of numbers of the form $\frac{m}{b^n}$ are assumed to end in zeros.

Since the range of E_b is finite, we have $E_b \preccurlyeq_S \mathcal{D}\text{iff}_b$.

In addition, $\mathcal{D} \preccurlyeq_S \mathcal{D}\text{iff}_b$. Given $q \in \mathbb{Q}$, we can compute the position of the first difference and then compare q and α using E_b.

Difference with respect to base-b expansions

As a first example we consider $\mathcal{D}\mathrm{iff}_b$, which corresponds to the base-b expansion E_b. The base-b expansions of numbers of the form $\frac{m}{b^n}$ are assumed to end in zeros.

Since the range of E_b is finite, we have $E_b \preccurlyeq_S \mathcal{D}\mathrm{iff}_b$.

In addition, $\mathcal{D} \preccurlyeq_S \mathcal{D}\mathrm{iff}_b$. Given $q \in \mathbb{Q}$, we can compute the position of the first difference and then compare q and α using E_b.

It is also easy to produce a trace function from below using $\mathcal{D}\mathrm{iff}_b$: $T^{\uparrow}(q) = \sum_{i=0}^{n} E_b(i) \cdot b^{-i}$, where $n = \mathcal{D}\mathrm{iff}_b(q)$.

Difference with respect to base-b expansions

As a first example we consider $\mathcal{D}\mathrm{iff}_b$, which corresponds to the base-b expansion E_b. The base-b expansions of numbers of the form $\frac{m}{b^n}$ are assumed to end in zeros.

Since the range of E_b is finite, we have $E_b \preccurlyeq_S \mathcal{D}\mathrm{iff}_b$.

In addition, $\mathcal{D} \preccurlyeq_S \mathcal{D}\mathrm{iff}_b$. Given $q \in \mathbb{Q}$, we can compute the position of the first difference and then compare q and α using E_b.

It is also easy to produce a trace function from below using $\mathcal{D}\mathrm{iff}_b$: $T^{\uparrow}(q) = \sum_{i=0}^{n} E_b(i) \cdot b^{-i}$, where $n = \mathcal{D}\mathrm{iff}_b(q)$.

In the paper [4] we prove that: $\hat{G} \prec_S \mathcal{D}\mathrm{iff}_b$ and also that $\check{G}$ and $\mathcal{D}\mathrm{iff}_b$ are subrecursively incomparable.

Difference with respect to base-b sum approximations

Now let us consider $\mathcal{D}\mathrm{iff}_{b\uparrow}$, which correponds to the base-b sum approximation from below $\hat{A}_b$.

Difference with respect to base-b sum approximations

Now let us consider $\mathcal{D}\mathrm{iff}_{b\uparrow}$, which correponds to the base-b sum approximation from below $\hat{A}_b$.
Observe now that the range of $\hat{A}_b$ is infinite.

Difference with respect to base-b sum approximations

Now let us consider $\mathcal{D}\mathrm{iff}_{b\uparrow}$, which correponds to the base-b sum approximation from below $\hat{A}_b$.
Observe now that the range of $\hat{A}_b$ is infinite.

We have $E_b \preccurlyeq_S \mathcal{D}\mathrm{iff}_{b\uparrow}$ with the following algorithm:
Given n and the digits $E_b(1), \ldots, E_b(n-1)$,

1. For each $\mathtt{D} \in \{1, \ldots, b-1\}$:
    1.1. Let $q = E_b(1) \cdot b^{-1} + \ldots + E_b(n-1) \cdot b^{-n+1} + x \cdot b^{-n}$.
    1.2. Let k be the number of non-zero base-b digits of q.
    1.3. If $\mathcal{D}\mathrm{iff}_{b\uparrow}(q) > k$ then output $E_b(n) = \mathtt{D}$.
2. Output $E_b(n) = \mathtt{0}$.

Difference with respect to base-b sum approximations

Now let us consider $\mathcal{D}\mathrm{iff}_{b\uparrow}$, which correponds to the base-b sum approximation from below $\hat{A}_b$.
Observe now that the range of $\hat{A}_b$ is infinite.

We have $E_b \preccurlyeq_S \mathcal{D}\mathrm{iff}_{b\uparrow}$ with the following algorithm:
Given n and the digits $E_b(1), \ldots, E_b(n-1)$,

1. For each $\mathtt{D} \in \{1, \ldots, b-1\}$:
    1.1. Let $q = E_b(1) \cdot b^{-1} + \ldots + E_b(n-1) \cdot b^{-n+1} + x \cdot b^{-n}$.
    1.2. Let k be the number of non-zero base-b digits of q.
    1.3. If $\mathcal{D}\mathrm{iff}_{b\uparrow}(q) > k$ then output $E_b(n) = \mathtt{D}$.
2. Output $E_b(n) = \mathtt{0}$.

Now we also have $\mathcal{D} \preccurlyeq_S \mathcal{D}\mathrm{iff}_{b\uparrow}$. Given $q \in \mathbb{Q}$, we can compare q and α using the base-b expansions: they differ in at least one position $\leq \ell$, where ℓ is the weight of $\hat{A}_b^q(\mathcal{D}\mathrm{iff}_{b\uparrow}(q))$.

Incomparability of $\mathcal{D}\mathrm{iff}_{b\uparrow}$ with trace functions

Theorem

$$\mathcal{D}\mathrm{iff}_{b\uparrow} \not\preceq_S \hat{G}, \;\; \mathcal{D}\mathrm{iff}_{b\uparrow} \not\preceq_S \check{G}$$

Incomparability of $\mathcal{D}\mathrm{iff}_{b\uparrow}$ with trace functions

Theorem

$$\mathcal{D}\mathrm{iff}_{b\uparrow} \not\preccurlyeq_S \hat{G}, \ \ \mathcal{D}\mathrm{iff}_{b\uparrow} \not\preccurlyeq_S \check{G}$$

Take the following irrational number:

$$\alpha = 0.000\ldots.000010101\ldots.D_0 0010101\ldots.D_1 00101\ldots..$$

$$\uparrow d_0 \qquad\qquad \uparrow d_1 \qquad\qquad \uparrow d_2$$

Incomparability of $\mathcal{D}\mathrm{iff}_{b\uparrow}$ with trace functions

Theorem

$$\mathcal{D}\mathrm{iff}_{b\uparrow} \not\preccurlyeq_S \hat{G}, \quad \mathcal{D}\mathrm{iff}_{b\uparrow} \not\preccurlyeq_S \check{G}$$

Take the following irrational number:

$$\alpha = 0.000\ldots.000010101\ldots.D_0 0010101\ldots.D_1 00101\ldots..$$

$$\uparrow d_0 \qquad \uparrow d_1 \qquad \uparrow d_2$$

Theorem

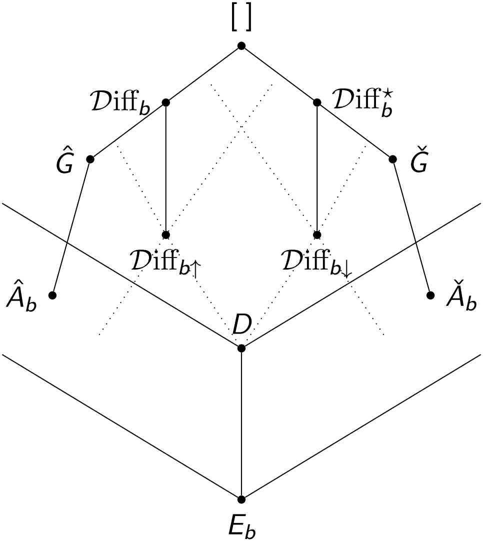
$$\hat{A}_b \not\preccurlyeq_S \mathcal{D}\mathrm{iff}_{b\uparrow}, \quad \check{A}_b \not\preccurlyeq_S \mathcal{D}\mathrm{iff}_{b\uparrow}$$

Take the following irrational number:

$$\alpha = 0.000\ldots.0001000\ldots.0001000\ldots.0001000\ldots..$$

$$\uparrow d_0 \qquad \uparrow d_1 \qquad \uparrow d_2$$

Picture of S-degrees



A. M. Ben-Amram and L. Kristiansen. *A Degree Structure on Representations of Irrational Numbers.*
Journal of Logic an Analysis, vol. 17 (2025), 1–21.

I. Georgiev.
Interplay between insertion of zeros and the complexity of Dedekind cuts.
Computability, vol. 13(2) (2024), 135–159.

I. Georgiev.
Subrecursive incomparability of the graphs of standard and dual Baire sequences.
Annuaire de l'Université de Sofia "St. Kliment Ohridski". Faculté de Mathématiques et Informatique, vol. 109 (2022), 41–55.

I. Georgiev and L. Kristiansen.
On $\mathcal{S}$-Degrees of Some Representations of Irrational Numbers.
Beckmann, A. and others (eds), Crossroads of Computability and Logic: Insights, Inspirations, and Innovations, CiE Proceedings 2025, LNCS, Springer, Cham, vol. 15764 (2025), 222–236.

I. Georgiev, L. Kristiansen and F. Stephan.
Computable irrational numbers with representations of surprising complexity.
Annals of Pure and Applied Logic, vol. 172(2) (2021), 102893.

K. Hiroshima and A. Kawamura.
Elementarily Traceable Irrational Numbers
In: Della Vedova, G. and others (eds), Unity of Logic and Computation, CiE Proceedings 2023, LNCS, Springer, Cham, vol. 13967 (2023), 135–140.

L. Kristiansen.
On subrecursive representability of irrational numbers, part II.
Computability, vol. 8(1) (2019), 43–65.

L. Kristiansen.
On subrecursive representability of irrational numbers.
Computability, vol. 6(3) (2017), 249–276.

Thanks for your attention!