Computable Analysis via Representations

Peter Hertling
Institut für Theoretische Informatik und Mathematik
Universität der Bundeswehr München, Germany

Overview

1. Computing over countable sets with notations
   - Notations, e.g., of natural numbers
   - Reducibility and equivalence
   - Relative computability
2. Computing over uncountable sets with representations
   - Representations, e.g., of real numbers
   - Reducibility and equivalence
   - Relative computability
   - The role of continuity
   - Admissible representations
   - Representations of functions
   - Computable metric spaces
3. Some results for illustration
   - A result by Pour-El and Richards
   - ... for the wave equation
   - Results by Weihrauch and Zhong for the wave equation

1 Computing over countable sets with notations
Notations, e.g., of natural numbers

Example

Given: a digital computer.
Task: perform some computation
involving natural numbers $n \in \{0, 1, 2, 3, \ldots\}$.

Necessary: one has to encode natural numbers via bits $0, 1$.
Possibilities:

- unary encoding: e.g. number $9 \cong$ encoded as 111111111.
- binary encoding: e.g. number $9 \cong$ encoded as 101.

Notations, e.g., of natural numbers

Now, let's be more formal.
In the following

- $f :\subseteq X \to Y$ means: f is a function
  - defined on some subset of X
  - with range in Y.
- Σ, Σ', etc. are finite, non-empty sets, i.e., alphabets, e.g., $\Sigma = \{0, 1\}$.
- Σ^* is the set of all finite strings over Σ.

Definition

A notation of a set X is a surjective function $\nu :\subseteq \Sigma^* \to X$.

(*surjective*
$=$ *onto*
$=$ for every $x \in X$ there is some $w \in \Sigma^*$ with $\nu(w) = x$)

Then any w with $\nu(w) = x$ is a ν-name for x.

Examples

- Unary notation: $\nu_{\text{unary}} :\subseteq \Sigma^* \to \mathbb{N}$ with

$$\nu_{\text{unary}}(1^n) := n.$$

- Binary notation: $\nu_{\text{binary}} :\subseteq \Sigma^* \to \mathbb{N}$ with

$$\nu_{\text{binary}}(a_k \dots a_0) := \begin{cases} \sum_{i=0}^{k} a_i \cdot 2^i & \text{if } a_k, \dots, a_0 \in \{0,1\},\ a_k \neq 0 \\ & \text{or if } k = 0 \text{ and } a_k = 0, \\ \text{undefined} & \text{otherwise} \end{cases}$$

Notations: Reducibility and Equivalence

Definition

Let $\nu_1 :\subseteq \Sigma^* \to X$ and $\nu_2 :\subseteq \Sigma'^* \to X$ be notations. Then

- ν_1 is reducible to ν_2, written $\nu_1 \leq \nu_2$, if there is a computable "translator" $T :\subseteq \Sigma^* \to \Sigma'^*$ with

  $$\nu_1(w) = \nu_2(T(w))$$

  for all $w \in \operatorname{dom} \nu_1$.
- ν_1 is equivalent to ν_2, written $\nu_1 \equiv \nu_2$, if $\nu_1 \leq \nu_2$ and $\nu_2 \leq \nu_1$.

Example

ν_{unary} and ν_{binary} are equivalent.

Notations: Complexity Considerations

The reducibility and equivalence relations just introduced are quite rough.
For practice also important: complexity considerations.

Example

ν_{unary} and ν_{binary} are *not* "polynomial time equivalent":
there is *no* translator from ν_{binary} to ν_{unary} that works in polynomial time!

Relative Computability: Functions

Definition

Let $\nu_X :\subseteq \Sigma^* \to X$ and $\nu_Y :\subseteq \Sigma'^* \to Y$ be notations. A function $f :\subseteq X \to Y$ is (ν_X, ν_Y)-computable or computable with respect to ν_X and ν_Y if there is a computable "realizer" $F :\subseteq \Sigma^* \to \Sigma'^*$ with

$$f\nu_X(w) = \nu_Y F(w)$$

for all $w \in \operatorname{dom} f\nu_X$.

Examples

- Addition of natural numbers: $+ : \mathbb{N}^2 \to \mathbb{N}$, $(n, m) \to n + m$, is $({\nu_{\text{binary}}}^2, \nu_{\text{binary}})$-computable, i.e., computable with respect to ν_{binary}.
- Addition of natural numbers is computable with respect ν_{unary} as well.

Here ${\nu_{\text{binary}}}^2$ is defined by

$${\nu_{\text{binary}}}^2(w_1 \# w_2) = (\nu_{\text{binary}}(w_1), \nu_{\text{binary}}(w_2))$$

if $w_1, w_2 \in \operatorname{dom} \nu_{\text{binary}}$.

Lemma

Let $\nu_X, \nu'_X :\subseteq \Sigma^* \to X$ and $\nu_Y, \nu'_Y :\subseteq \Sigma'^* \to Y$ be notations.
If $\nu_X \equiv \nu'_X$ and $\nu_Y \equiv \nu'_y$, then for any function $f :\subseteq X \to Y$:

$$f \text{ is } (\nu_X, \nu_Y)\text{-computable} \iff f \text{ is } (\nu'_X, \nu'_Y)\text{-computable.}$$

Relative Computability: Sets

For subsets $A \subseteq \mathbb{N}$ important notions:

- computability (decidability)
- computable enumerability (recursive enumerability).

They can also be relativised in a natural way.

(Omitted).

Notations: Another Example

Example

A notation $\nu_{\mathbb{Q}}$ of rational numbers can be defined by:

$$\nu_{\mathbb{Q}}(s\, w_1 \# w_2) = s \frac{\nu_{\text{binary}}(w_1)}{\nu_{\text{binary}}(w_2)}$$

if $s \in \{+, -\}$, $w_1, w_2 \in \operatorname{dom} \nu_{\text{binary}}$, $\nu_{\text{binary}}(w_2) \neq 0$.

Notations: Three Remarks

Remarks

1. For all countable sets over which one usually performs computations, a natural choice of a notation is usually "good".
2. More care is required if complexity is also an issue (in practice always).
3. For many structures ($=$ sets with operations on them) the wish to perform the operations effectively already determines which notation one should use, up to equivalence.

All this applies, e.g., to $\mathbb{N}$ and $\mathbb{Q}$. Therefore, we fix "good" notations for these sets and simply say that we are *computing with natural and rational numbers*.

2 Computing over uncountable sets with representations

Example

Given: a digital computer.
Task: perform some computation
involving real numbers $r \in \mathbb{R}$.

Necessary: one has to encode real numbers via bits $0, 1$.
Problem:
There are only countably many binary strings,
but there are uncountably many real numbers!
Idea:
Encode real numbers by *infinite* binary strings!

Representations, e.g., of Real Numbers

$\Sigma^\omega := \{p \mid p : \mathbb{N} \to \Sigma\}$ = set of one-way infinite sequences over Σ.

Definition

A representation of a set X is a surjective function $\rho :\subseteq \Sigma^\omega \to X$.
Then any w with $\rho(w) = x$ is a ρ-name for x.

Examples

- Decimal representation: defined in the usual way, e.g., $\rho_{\text{decimal}}(0.5000\ldots) = 1/2$, $\rho_{\text{decimal}}(-3.1415927\ldots) = -\pi$, ...
- Representation via rational intervals:

$$
\begin{aligned}
\rho_{\text{interval}}(p) = x \iff \quad & p = a_0\#b_0\#a_1\#b_1\#a_2\#b_2\#\ldots \\
& \text{and } \nu_{\mathbb{Q}}(a_0) < \nu_{\mathbb{Q}}(a_1) < \nu_{\mathbb{Q}}(a_2) < \ldots < x \\
& \qquad < \ldots < \nu_{\mathbb{Q}}(b_2) < \nu_{\mathbb{Q}}(b_1) < \nu_{\mathbb{Q}}(b_0) \\
& \text{and } \lim_{n\to\infty} \nu_{\mathbb{Q}}(a_n) = x = \lim_{n\to\infty} \nu_{\mathbb{Q}}(b_n).
\end{aligned}
$$

More Representations of Real Numbers

We say that p *encodes a rational sequence* $a_0, a_1, a_2, \ldots$ if $p = w_0 \# w_1 \# w_2 \# \ldots$ with $\nu_{\mathbb{Q}}(w_i) = a_i$ for all i.

Examples

$$\rho_{\text{naiveCauchy}}(p) = x \iff p \text{ encodes a rational sequence } a_0, a_1, a_2, \ldots \text{ with } \lim_{n\to\infty} a_n = x.$$

$$\rho_{\text{normedCauchy}}(p) = x \iff p \text{ encodes a rational sequence } a_0, a_1, a_2, \ldots \text{ with } \lim_{n\to\infty} a_n = x \text{ and } |a_n - a_m| \leq 2^{-\min\{m,n\}} \text{ for all } n, m.$$

$$\rho_{\text{increasing}}(p) = x \iff p \text{ encodes a rational sequence } a_0, a_1, a_2, \ldots \text{ with } \lim_{n\to\infty} a_n = x \text{ and } a_0 < a_1 < a_2 < \ldots.$$

$$\rho_{\text{decreasing}}(p) = x \iff p \text{ encodes a rational sequence } a_0, a_1, a_2, \ldots \text{ with } \lim_{n\to\infty} a_n = x \text{ and } \ldots < a_2 < a_1 < a_0.$$

Which of these representations are useful?

Take care to choose a representation so that, using a digital computer, you can perform useful computations on these infinite descriptions of real numbers!

Relative Computability with Respect to Representations

Let $\rho :\subseteq \Sigma^\omega \to X$ and $\sigma :\subseteq \Sigma'^\omega \to Y$ be representations.

We want useful notions:

1. ρ-computable elements of X,
2. (ρ, σ)-computable (multi-valued) functions from X to Y,
3. computability notions for subsets of X, relative to ρ (this will be omitted).

Need: Useful computability notions for

1. $p \in \Sigma^\omega$,
2. $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$,
3. subsets of Σ^ω (omitted).

Computable elements

A sequence $p \in \Sigma^\omega$ is computable if either of the two following equivalent conditions is fulfilled:

- There is a Turing machine that, given any n, computes $p(n)$.
- There is a Turing machine that outputs $p(0)$, $p(1)$, $p(2)$, and so on, without ever halting.

Definition

Let $\rho_X :\subseteq \Sigma^\omega \to X$ be a representation. An element $x \in X$ is ρ-computable if there is a computable $p \in \Sigma^\omega$ with $\rho_X(p) = x$.

Relations between Computability Notions for Real Numbers

An arrow from ρ to σ means: ρ-computability implies σ-computability.

$$
\begin{tikzcd}
 & \rho_{\mathsf{naiveCauchy}} & \\
\rho_{\mathsf{increasing}} \arrow[ur] & & \rho_{\mathsf{decreasing}} \arrow[ul] \\
 & \rho_{\mathsf{normedCauchy}} \arrow[ul] \arrow[ur] \arrow[d, leftrightarrow] & \\
 & \rho_{\mathsf{interval}} \arrow[d, leftrightarrow] & \\
 & \rho_{\mathsf{decimal}} &
\end{tikzcd}
$$

Computable Real Numbers

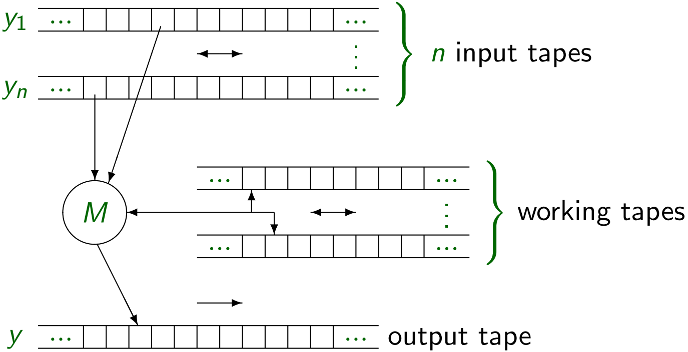
Definition

A real number is called computable if it is ρ_{interval}-computable.
$\mathbb{R}_c :=$ the set of computable real numbers.

Theorem

$\mathbb{R}_c$ is a field, real-algebraically closed, and closed under "effective" limit.

Computable Functions on Infinite Strings
Turing machine



$f_M(y_1, ..., y_n) = y$

Computable Functions on Infinite Strings

Definition

A function $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ is computable if there exists a Turing machine M which on input $p \in \Sigma^\omega$ behaves as follows:

- if $p \in \mathrm{dom}\, F$, then M writes $F(p)(0)$, $F(p)(1)$, $F(p)(2)$, $\ldots$ step by step on the output tape without ever going backwards on the output tape.
- if $p \notin \mathrm{dom}\, F$, the M does not produce an infinite output.

Remark

Note that each output bit $F(p)(i)$ must have been written after finitely many steps.
And until then, M can have read only finitely many input bits $p(0)$, $p(1)$, $p(2)$, $\ldots$.

Representations: Reducibility and Equivalence

Definition

Let $\rho_1 :\subseteq \Sigma^\omega \to X$ and $\rho_2 :\subseteq \Sigma'^\omega \to X$ be representations. Then

- ρ_1 is reducible to ρ_2, written $\rho_1 \le \rho_2$, if there is a computable "translator" $T :\subseteq \Sigma^\omega \to \Sigma'^\omega$ with

$$\rho_1(p) = \rho_2(T(p))$$

  for all $p \in \operatorname{dom} \rho_1$.
- ρ_1 is equivalent to ρ_2, written $\rho_1 \equiv \rho_2$, if $\rho_1 \le \rho_2$ and $\rho_2 \le \rho_1$.

Example

ρ_{interval} and $\rho_{\text{normedCauchy}}$ are equivalent.
ρ_{interval} and ρ_{decimal} are not equivalent.

Relations between Real Number Representations

An arrow from ρ to σ means: $\rho \leq \sigma$.

$$
\begin{tikzcd}
& \rho_{\mathsf{naiveCauchy}} & \\
\rho_{\mathsf{increasing}} \arrow[ur] & & \rho_{\mathsf{decreasing}} \arrow[ul] \\
& \rho_{\mathsf{normedCauchy}} \arrow[ul] \arrow[ur] \arrow[d, leftrightarrow] & \\
& \rho_{\mathsf{interval}} & \\
& \rho_{\mathsf{decimal}} \arrow[u] &
\end{tikzcd}
$$

Reducibility and Computable Elements

Lemma

1. *If $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ is computable and $p \in \mathrm{dom}\, F$ is computable, then also $F(p)$ is computable.*
2. *If $\rho_1 \leq \rho_2$, then for $x \in X$: ρ_1-computable $\Rightarrow$ ρ_2-computable.*
3. *If $\rho_1 \equiv \rho_2$, then for $x \in X$: ρ_1-computable $\iff$ ρ_2-computable.*

The inverse of the 2nd statement is not true!

Example

On the one hand:

$\rho_{\text{decimal}} \leq \rho_{\text{interval}}$, but $\rho_{\text{interval}} \not\leq \rho_{\text{decimal}}$.
On the other hand for real numbers:

ρ_{decimal}-computable $\iff$ ρ_{interval}-computable!

Relative Computability: Functions

Definition

Let $\rho_X :\subseteq \Sigma^\omega \to X$ and $\rho_Y :\subseteq \Sigma'^\omega \to Y$ be representations. A function $f :\subseteq X \to Y$ is (ρ_X, ρ_Y)-computable or computable with respect to ρ_X and ρ_Y if there is a computable "realizer" $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ with

$$f\nu_X(p) = \nu_Y F(p)$$

for all $p \in \operatorname{dom} f\nu_X$.

Lemma

Let $\rho_X, \rho'_X :\subseteq \Sigma^\omega \to X$ and $\rho_Y, \rho'_Y :\subseteq \Sigma'^\omega \to Y$ be representations. If $\rho_X \equiv \rho'_X$ and $\rho_Y \equiv \rho'_y$, then for any function $f :\subseteq X \to Y$:

$$f \text{ is } (\rho_X, \rho_Y)\text{-computable} \iff f \text{ is } (\rho'_X, \rho'_Y)\text{-computable.}$$

Relative Computability w.r.t. ρ_{decimal}

For a representation ρ, define ρ^2 by

$$\rho^2(p) := (\rho(p(0)p(2)p(4)\ldots), \rho(p(1)p(3)p(5)\ldots)).$$

Is addition on real numbers: $+ : \mathbb{R}^2 \to \mathbb{R}$, $(x, y) \to x + y$ a $(\rho^2_{\text{decimal}}, \rho_{\text{decimal}})$-computable function?

Let us look at input $.44444\ldots$ and $.55555\ldots$.
How should the output start: $.9$ or 1.0?
Impossible to decide after reading only finitely many input digits! So, addition is *not* computable w.r.t. ρ_{decimal}!

Similar observation for multiplication.

Relative Computability w.r.t. ρ_{interval}

Theorem

The following functions over the real numbers are computable with respect to $\rho_{interval}$.

- $+, -, *, / :\subseteq \mathbb{R}^2 \to \mathbb{R}$,
- $x \to |x|$ *and* $\min, \max : \mathbb{R}^2 \to \mathbb{R}$,
- *the constant function* $x \to c$ *if* c *is a computable real number,*
- $\exp$, $\sin$, $\cos$, $\log$, $\sqrt{\ }$.

The representations equivalent to ρ_{interval} are the most useful representations of $\mathbb{R}$
(w.r.t to computability; w.r.t complexity one still has to be more selective).

We call a real function computable if it is computable with respect to ρ_{interval}.

The Role of Continuity

Lemma

The function $d : \Sigma^\omega \times \Sigma^\omega \to \mathbb{R}$ *defined by*

$$d(p,q) := \begin{cases} 2^{-\min\{i \mid p(i) \neq q(i)\}} & \textit{if } p \neq q \\ 0 & \textit{otherwise} \end{cases}$$

is a metric on Σ^ω.

Lemma

The representation $\rho_{interval}$ *is continuous,*
as is every representation equivalent to it.
Furthermore, all representations equivalent to it have an open and surjective restriction.

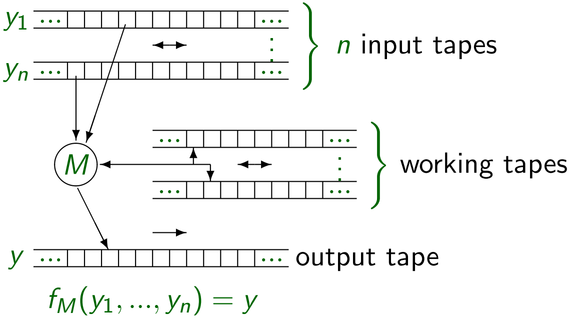
Continuity of Computable Functions on Strings

Theorem

Every computable function $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ is continuous.

Proof.

Remember that each output bit $F(p)(i)$ must have been written after finitely many steps. And until then, M can have read only finitely many input bits $p(0)$, $p(1)$, $p(2)$, $\ldots$. □



Continuity of Computable Functions over the Real Numbers

Theorem

Every (w.r.t. ρ_{interval}) computable function $f :\subseteq \mathbb{R}^n \to \mathbb{R}$ is continuous.

Proof.

Consider $n = 1$ and some TM computing a realizer for f.
Fix some $x \in \operatorname{dom} f$ and some $\varepsilon > 0$.
Let p be a ρ_{interval}-name of x. After finitely many steps, the TM must have produced an output interval J with length $< \varepsilon$.
But during these finitely many steps, the TM has read only a finite prefix of p. This prefix is also the prefix of ρ_{interval}-names of all real numbers y in some open interval I containing x.
Hence, $y \in I \implies f(y) \in J$.
That means: f is continuous. □

Admissible Representations

Definition

A representation ρ of a topological space X is called admissible if every representation σ of X is *continuously reducible* to ρ, i.e. there exists a continuous "translator" T with $\rho(p) = \sigma T(p)$ for all $p \in \operatorname{dom} \rho$.

Theorem (Kreitz, Weihrauch)

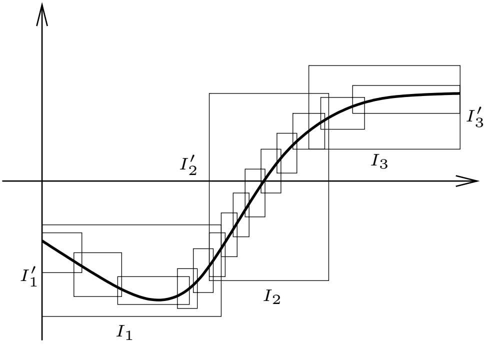
Let $\rho_X :\subseteq \Sigma^\omega \to X$ and $\rho_Y :\subseteq \Sigma'^\omega \to Y$ be admissible representations of T_0-spaces with countable base. Then a function $f :\subseteq X \to Y$ is continuous if, and only if, f is (ρ_X, ρ_Y)-continuous, i.e. there exists a continuous realizer $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ with $f\rho_X(p) = \rho_Y F(p)$ for all $p \in \operatorname{dom} f\rho_X$.

Remark

Further generalised by Schröder to admissibly represented "weak limit spaces": there consider *sequential continuity*.

Representation of Continuous Real Functions f

$\rho_{\mathsf{cont}}(p) = f \iff p$ enumerates a list of pairs of open rational intervals (I_i, I_i') with $f(\mathrm{closure}(I_i)) \subseteq I_i'$ and such that for any $x \in \mathrm{dom}(f)$ there exist arbitrarily small I_i' with $x \in I_i$.



In a similar way a one can define a representation of continuous functions $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ (with G_δ-domains).

Representations of Continuous Functions

From a suitable representation of (certain) continuous functions $F :\subseteq \Sigma^\omega \to \Sigma'^\omega$ one obtains:

Theorem

Let $\rho_X :\subseteq \Sigma^\omega \to X$ and $\rho_Y :\subseteq \Sigma'^\omega \to Y$ be representations. Then there exist representations $[\rho_X, \rho_Y]$ of $X \times Y$ and $[\rho_X \to \rho_Y]$ of the space of (ρ_X, ρ_Y)-continuous functions with the following properties:

- *(evaluation) the function*

  $$(f, x) \to f(x)$$

  is $[[\rho_X \to \rho_Y], \rho_X], \rho_Y)$-computable,
- *(type conversion) any function $f : Z \times X \to Y$ is $[\rho_Z, \rho_X], \rho_Y)$-computable if, and only if, the function*

  $$z \to (x \to f(z, y))$$

  is $(\rho_Z, [\rho_X \to \rho_Y])$-computable.

Representations of Continuous Functions

Theorem

Let $\rho_X :\subseteq \Sigma^\omega \to X$ *and* $\rho_Y :\subseteq \Sigma'^\omega \to Y$ *be representations. For a function* $f : X \to Y$ *the following conditions are equivalent:*

- f *is* (ρ_X, ρ_Y)*-computable.*
- f *is a* $[\rho_X \to \rho_Y]$*-computable element of the space of* (ρ_X, ρ_Y)*-continuous functions.*

Computable Metric Space

A triple (X, d, α) is a computable metric space if

1. $d : X \times X \to \mathbb{R}$ is a metric on the set X,
2. $\alpha : \mathbb{N} \to X$ is a sequence dense in X,
3. $d \circ (\alpha \times \alpha) : \mathbb{N}^2 \to \mathbb{R}$ is a computable double sequence of real numbers.

Example

$(\mathbb{R}, |\cdot|, \nu_{\mathbb{Q}})$, where $\nu_{\mathbb{Q}} : \mathbb{N} \to \mathbb{Q}$ is some standard numbering of the rational numbers.

Example

$(C[a, b], (f, g) \to ||f - g||_\infty, \alpha)$, where $a < b$ are computable real numbers and $\alpha : \mathbb{N} \to C[a, b]$ is some standard numbering of spline functions on a, b with rational breakpoints.

Computable Metric Space: a Representation

Definition

Let (X, d, α) be a computable metric space. Then the representation $\rho_{X,\text{normedCauchy}}$ of X is defined as for $\mathbb{R}$:

$$\rho_{X,\text{normedCauchy}}(p) = x \iff \begin{array}{l} p = w_0 \# w_1 \# w_2 \# \ldots \\ \text{and } \lim_{n\to\infty} \alpha(w_n) = x \\ \text{and } d(\alpha(w_n), \alpha(w_m)) \leq 2^{-\min\{n,m\}} \\ \text{for all } n, m. \end{array}$$

Let $a < b$ be computable real numbers.

Lemma

Consider the computable metric space $(C[a,b], (g,h) \to ||g-h||_\infty, \alpha)$. Its two representations $[\rho_{interval}|^{[a,b]} \to \rho_{interval}]$ and $\rho_{C[a,b],normedCauchy}$ are equivalent.

Corollary

For a function $f : [a,b] \to \mathbb{R}$ the following conditions are equivalent:

- f *is computable w.r.t.* $\rho_{interval}$ *resp. its restriction to names of* $[a,b]$.
- f *is a* $[\rho_{interval}|^{[a,b]} \to \rho_{interval}]$*-computable element of* $C[a,b]$.
- f *is a* $\rho_{C[a,b],normedCauchy}$*-computable element of* $C[a,b]$.

3 Some Results for Illustration
Computable normed space, computable Banach space

In the following we assume that $(\mathbb{F}, d, \alpha_{\mathbb{F}})$ is either $(\mathbb{R}, |\cdot|, \nu_{\mathbb{Q}})$ or $(\mathbb{C}, d, \nu_{\mathbb{Q}[i]})$.

A tuple $(X, ||\cdot||, e)$ is a computable normed space over $\mathbb{F}$ if $(X, ||\cdot||)$ is a normed linear space over $\mathbb{F}$ with

1. e is a fundamental sequence, i.e., its linear span is dense in X,
2. (X, d, α_e) with $d(x, y) := ||x - y||$ and $\alpha_e\langle k, \langle n_1, \ldots, n_k\rangle\rangle := \sum_{i=1}^{k} \alpha_{\mathbb{F}}(n_i) \cdot e_i$ is a computable metric space,
3. 0 is a computable element,
   $\cdot : \mathbb{F} \times X \to X$, $(a, x) \mapsto a \cdot x$ is computable,
   $+ : X \times X \to X$, $(x, y) \mapsto x + y$ is computable

If such a space is complete then it is a computable Banach space.

First Main Theorem of Pour-El and Richards

Theorem

Let $(X, || \cdot ||, e)$ *and* Y *be computable Banach spaces,*
$T : \mathrm{dom}(T) \subseteq X \to Y$ *a closed linear operator with*
$\{e_n \mid n \in \mathbb{N}\} \subseteq \mathrm{dom}(T)$ *and*
such that the sequence $(T(e_n))_n$ *is a computable sequence in* Y*.*
Then

1. *if* T *is bounded, then* T *preserves computability,*
2. *if* T *is unbounded, then* T *does not preserve computability, i.e., there is some computable* $x \in \mathrm{dom}(f)$ *such that* $f(x) \in Y$ *is not computable.*

Remark

Claim 1 can be strengthened: "$\ldots$, then T is computable w.r.t. the normed Cauchy representations".
Claim 2 is a stronger than the (trivial) claim: "$\ldots$, then T is not computable w.r.t. the normed Cauchy representations".

Negative result for the wave equation

Three-dimensional wave equation:

$$\begin{cases} u_{tt} = \Delta u, \\ u(0,x) = f(x), \quad u_t(0,x) = 0, \quad t \in \mathbb{R},\ x \in \mathbb{R}^3. \end{cases} \tag{1}$$

For $f \in C^1(\mathbb{R}^3)$ there is a unique solution $u \in C^0(\mathbb{R}^4)$.

Corollary (Pour-El and Richards)

There exists a **computable** *function* $f = u(0,\cdot) : \mathbb{R}^3 \to \mathbb{R}$ *with* $f \in C^1(\mathbb{R}^3)$ *such that the function* $u(1,\cdot) : \mathbb{R}^3 \to \mathbb{R}$ *is* **not computable**.

Reason for the negative result

Three-dimensional wave equation:

$$\begin{cases} u_{tt} = \Delta u, \\ u(0,x) = f(x), \quad u_t(0,x) = g(x), \quad t \in \mathbb{R},\ x \in \mathbb{R}^3. \end{cases} \tag{2}$$

For $f \in C^1(\mathbb{R}^3)$ and $g \in C^0(\mathbb{R}^3)$ there is a unique solution $u \in C^0(\mathbb{R}^4)$ given by

$$u(t,x) = \int_{\text{unit sphere}} (tg(x+tn) + f(x+tn) + t(\operatorname{grad} f)(x+tn) d\sigma(n).$$

Derivative causes loss of one degree of smoothness
and causes unboundedness of the operator with respect to the $||\cdot||_\infty$-norm.

Positive result for the wave equation

Theorem (Weihrauch, Zhong 2002)

Let $k \geq 1$. *The solution operator* $S : C^k(\mathbb{R}^3) \times C^{k-1}(\mathbb{R}^3) \to C^{k-1}(\mathbb{R}^4)$ *mapping* (f, g) *to the solution* u *is computable (with respect to suitable representations of the spaces* $C^k(\mathbb{R}^3)$, $C^{k-1}(\mathbb{R}^)$*!)*

Corollary (Pour-El/Richards, Weihrauch/Zhong 2002)

If f **and** f' *are computable and* g *is computable, then* u *is computable.*

Note:

Theorem (Myhill 1971)

There exists a computable function $f \in C^1(\mathbb{R})$ *such that its derivative* f' *is not computable.*

Another positive result for the wave equation

Theorem (Weihrauch, Zhong 2002)

Let $s \in \mathbb{R}$. The solution operator

$$\begin{aligned} S : H^s(\mathbb{R}^3) \times H^{s-1}(\mathbb{R}^3) \times \mathbb{R} &\to H^s(\mathbb{R}^3) \times H^{k-1}(\mathbb{R}^4) \\ (f, g, t) &\to (u(t, \cdot), u'(t, \cdot)) \end{aligned}$$

is computable (with respect to the normed Cauchy representations induced by the respective norms of these Sobolev spaces!).

Not treated

- Computability notions for sets of real numbers.
- The importance of multi-valued functions.
- More about topological aspects of representations
- Representations and complexity theory.
- The relation of other computability notions for real number functions to the notion explained here (computability w.r.t. ρ_{interval}).

References

This talk presented the approach to computable analysis worked out in:

Weihrauch, Klaus: *Computable Analysis. An Introduction.* Springer, 2000.

Web

`http://cca-net.de`

Conclusion

Representation approach to computable analysis

- is a rather concrete approach to computability over the reals,
- stresses that one should not loosely speak about

  "computing with mathematical objects"

  but rather about

  "computing with *information about* mathematical objects"