Analytic Functions and Small Complexity Classes

Akitoshi Kawamura, Florian Steinberg, **Holger Thies**

Workshop on Mathematical Logic and its Applications
Kyoto University, Kyoto, Japan

September 16, 2016

TTE: Computable Real Numbers

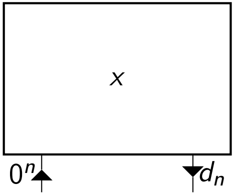
Computable Real Number

A real number is called computable if it can be approximated up to any desired precision.

TTE: Computable Real Numbers

Computable Real Number

A real number is called computable if it can be approximated up to any desired precision.



d_n rational approximations

$$|d_n - x| \leq 2^{-n}.$$

TTE: Computable Real Functions

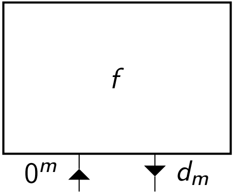
Computable Real Function

A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.

TTE: Computable Real Functions

Computable Real Function

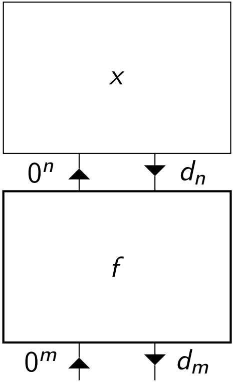
A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.



TTE: Computable Real Functions

Computable Real Function

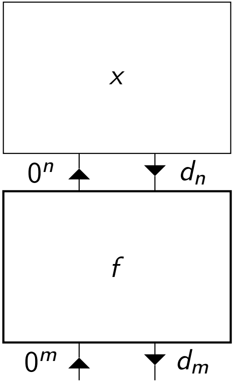
A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.



TTE: Computable Real Functions

Computable Real Function

A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.



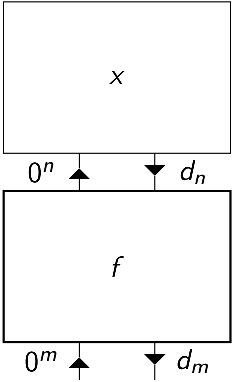
Notes

- closed under composition

TTE: Computable Real Functions

Computable Real Function

A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.



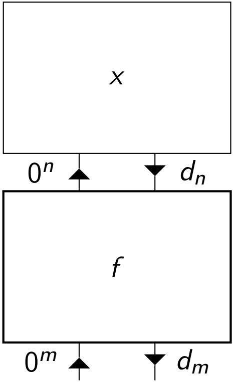
Notes

- closed under composition
- multidimensional functions: several input oracles and outputs.

TTE: Computable Real Functions

Computable Real Function

A function $f : [0,1] \to \mathbb{R}$ is called computable if the values $f(x)$ can be approximated up to any desired precision.



Notes

- closed under composition
- multidimensional functions: several input oracles and outputs.
- Computable functions are continuous.

Generalization

Representation

A representation for a set X is a partial surjective function $\rho : \mathcal{B} \to X$, that is, objects are encoded by string functions.

Generalization

Representation

A representation for a set X is a partial surjective function $\rho : \mathcal{B} \to X$, that is, objects are encoded by string functions.

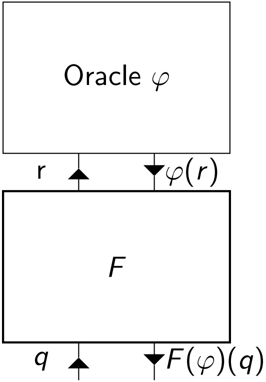
$$\begin{tikzcd}
\mathcal{B} \arrow[r, "F"] \arrow[d, "\alpha"'] & \mathcal{B} \arrow[d, "\beta"] \\
X \arrow[r, "f"'] & Y
\end{tikzcd}$$

Generalization

Representation

A representation for a set X is a partial surjective function $\rho : \mathcal{B} \to X$, that is, objects are encoded by string functions.

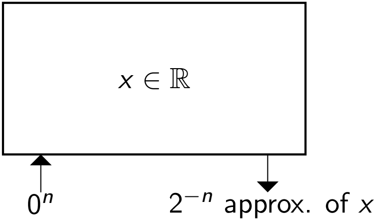
$$\begin{tikzcd}
\mathcal{B} \arrow[r, "F"] \arrow[d, "\alpha"'] & \mathcal{B} \arrow[d, "\beta"] \\
X \arrow[r, "f"'] & Y
\end{tikzcd}$$



Representation for Real Numbers

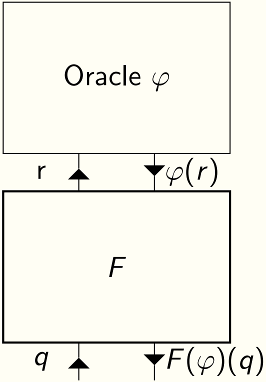
Example

Define a $\rho_{\mathbb{R}}$-name of $x \in \mathbb{R}$ by letting $\rho_{\mathbb{R}}(0^n)$ encode some $d \in \mathbb{D}$ such that $|d - x| \leq 2^{-n}$.

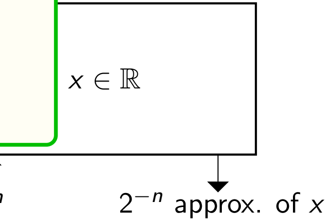


Numbers

$(\rho_{\mathbb{R}}^{|[0,1]}, \rho_{\mathbb{R}})$-computable

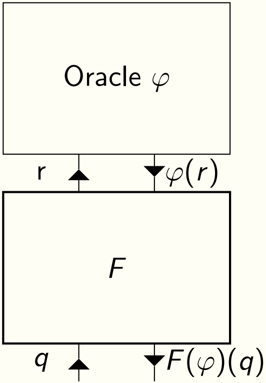


$\mathbb{R}$ by letting $\rho_{\mathbb{R}}(0^n)$ encode some $d \in \mathbb{D}$



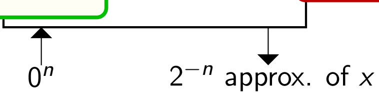
Numbers

$(\rho_{\mathbb{R}}^{|[0,1]}, \rho_{\mathbb{R}})$-computable

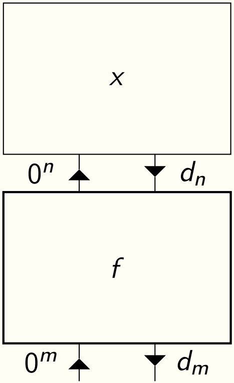


$\mathbb{R}$ by letting $\rho_{\mathbb{R}}$

$x \in \mathbb{R}$



Computable Function



Representation for Functions

Example

Define a $\delta_{\Box}$-name of a function $f \in C[0,1]$ as a pair $< \mu, \phi >$ where

- ϕ is such that $|\phi(d, 0^n) - f(d)| \leq 2^{-n}$ for all $d \in \mathbb{D} \cap [0,1]$
- μ encodes the modulus of continuity (in unary)

Representation for Fun

Modulus of Continuity

$$|x - y| \leq 2^{-\mu(n)} \Rightarrow |f(x) - f(y)| \leq 2^{-n}$$

Example

Define a $\delta_{\Box}$-name of a function $f \in C[0,1]$ as a pair $<\mu, \phi>$ where

- ϕ is such that $|\phi(d, 0^n) - f(d)| \leq 2^{-n}$ for all $d \in \mathbb{D} \cap [0,1]$
- μ encodes the modulus of continuity (in unary)

Representation for Fun

Modulus of Continuity

$$|x - y| \leq 2^{-\mu(n)} \Rightarrow |f(x) - f(y)| \leq 2^{-n}$$

Example

Define a $\delta_{\square}$-name of a function $f \in C[0,1]$ as a pair $< \mu, \phi >$ where

- ϕ is such that $|\phi(d, 0^n) - f(d)| \leq 2^{-n}$ for all $d \in \mathbb{D} \cap [0,1]$
- μ encodes the modulus of continuity (in unary)

$$\boxed{f \in C[0,1]}$$

Representation for Fun

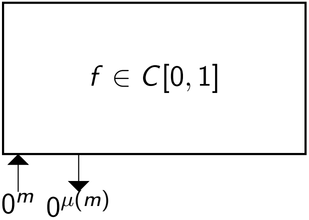
Modulus of Continuity

$$|x - y| \leq 2^{-\mu(n)} \Rightarrow |f(x) - f(y)| \leq 2^{-n}$$

Example

Define a $\delta_{\square}$-name of a function $f \in C[0,1]$ as a pair $< \mu, \phi >$ where

- ϕ is such that $|\phi(d, 0^n) - f(d)| \leq 2^{-n}$ for all $d \in \mathbb{D} \cap [0,1]$
- μ encodes the modulus of continuity (in unary)



Representation for Fun

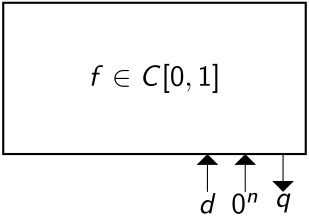
Modulus of Continuity

$$|x - y| \leq 2^{-\mu(n)} \Rightarrow |f(x) - f(y)| \leq 2^{-n}$$

Example

Define a $\delta_\square$-name of a function $f \in C[0,1]$ as a pair $< \mu, \phi >$ where

- ϕ is such that $|\phi(d, 0^n) - f(d)| \leq 2^{-n}$ for all $d \in \mathbb{D} \cap [0,1]$
- μ encodes the modulus of continuity (in unary)



Type-2 Complexity Theory

- Let Σ^{**} be the set of length-monotone string-functions

Type-2 Complexity Theory

- Let Σ^{**} be the set of length-monotone string-functions
- Define $|\varphi| : \mathbb{N} \to \mathbb{N}$ by $|\varphi|(|u|) = |\varphi(u)|$

Type-2 Complexity Theory

- Let Σ^{**} be the set of length-monotone string-functions
- Define $|\varphi| : \mathbb{N} \to \mathbb{N}$ by $|\varphi|(|u|) = |\varphi(u)|$
- Bound running time by second order polynomials $P(|\varphi|)(|x|)$

Type-2

Second Order Polynomials

Defined inductively by

1. $P(L,n) := m$, for $m \in \mathbb{N}$
2. $P(L,n) := n$
3. P, Q second order polynomials $\Rightarrow P + Q,\ P \cdot Q$
4. P a second order polynomial $\Rightarrow L(P)$

Example: $P(L,n) = 2L(L(L(n) \cdot L(n) + 2L(n)) + 2) + n + 10$

- Let
- Defi
- Bou

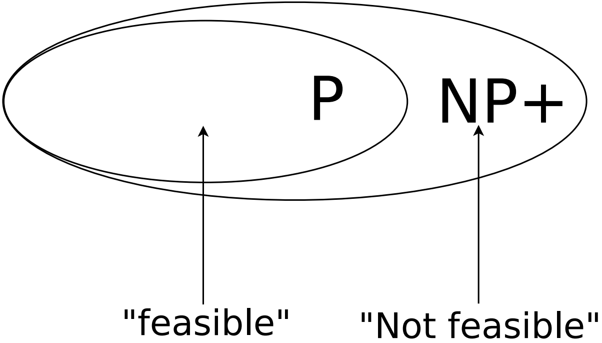
Type-2 Complexity Theory

- Let Σ^{**} be the set of length-monotone string-functions
- Define $|\varphi| : \mathbb{N} \to \mathbb{N}$ by $|\varphi|(|u|) = |\varphi(u)|$
- Bound running time by second order polynomials $P(|\varphi|)(|x|)$
- Can define complexity classes FP^2, $\#\mathrm{P}^2$ and $\mathrm{FPSPACE}^2$

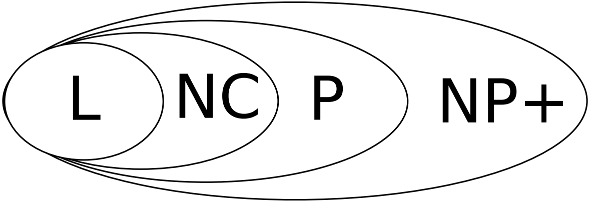
Type-2 Complexity Theory

- Let Σ^{**} be the set of length-monotone string-functions
- Define $|\varphi| : \mathbb{N} \to \mathbb{N}$ by $|\varphi|(|u|) = |\varphi(u)|$
- Bound running time by second order polynomials $P(|\varphi|)(|x|)$
- Can define complexity classes FP^2, $\#\mathrm{P}^2$ and $\mathrm{FPSPACE}^2$
- Can also define complexity classes P^2, NP^2 and PSPACE^2 by considering functions $\varphi : \Sigma^{**} \to (\Sigma^* \to \{0, 1\})$

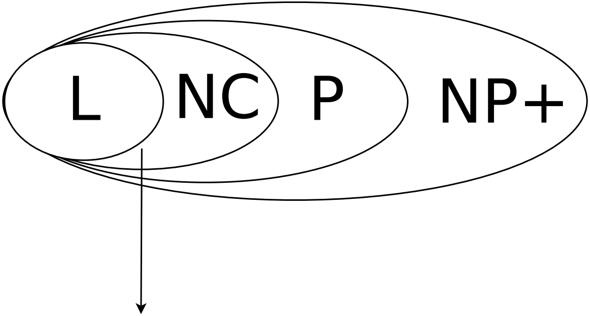
Small Complexity Classes



Small Complexity Classes

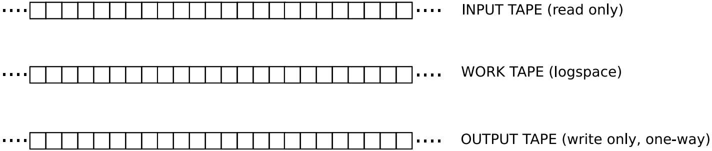


Small Complexity Classes



"efficiently solvable by a parallel computer"

Small Complexity Classes

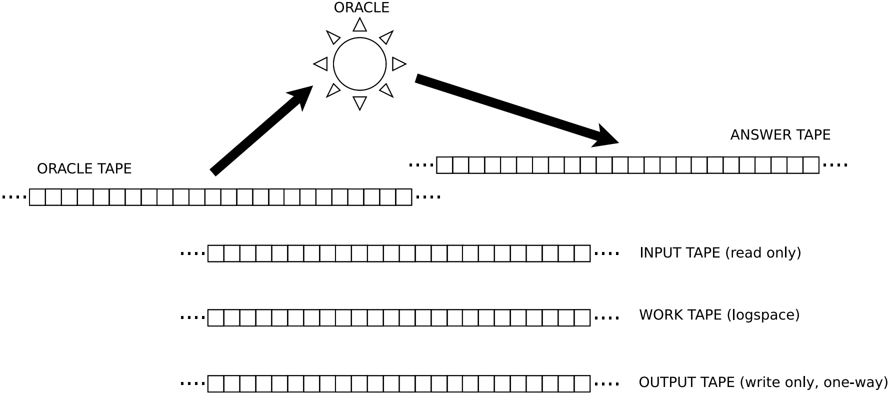


INPUT TAPE (read only)

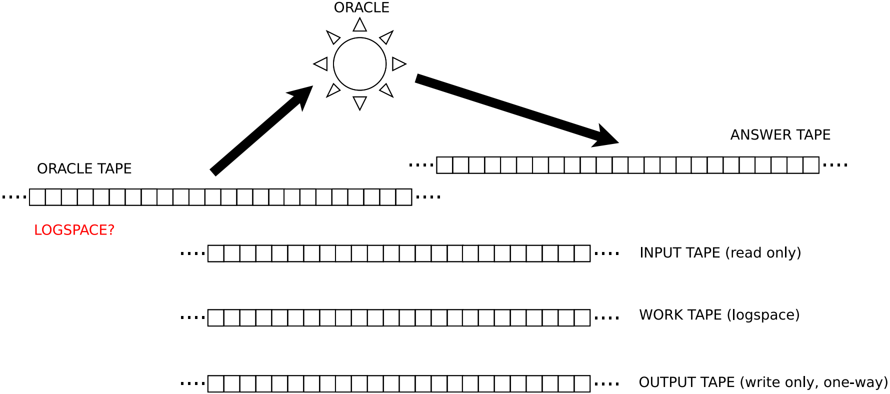
WORK TAPE (logspace)

OUTPUT TAPE (write only, one-way)

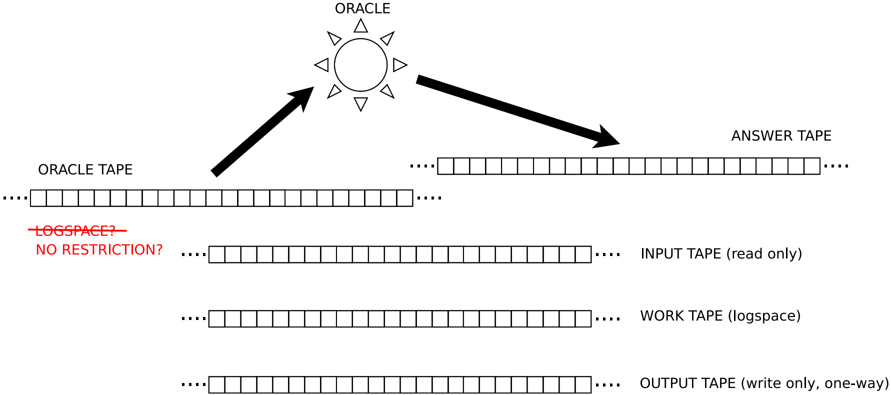
Small Complexity Classes



Small Complexity Classes



Small Complexity Classes



ORACLE

ANSWER TAPE

ORACLE TAPE

~~LOGSPACE?~~

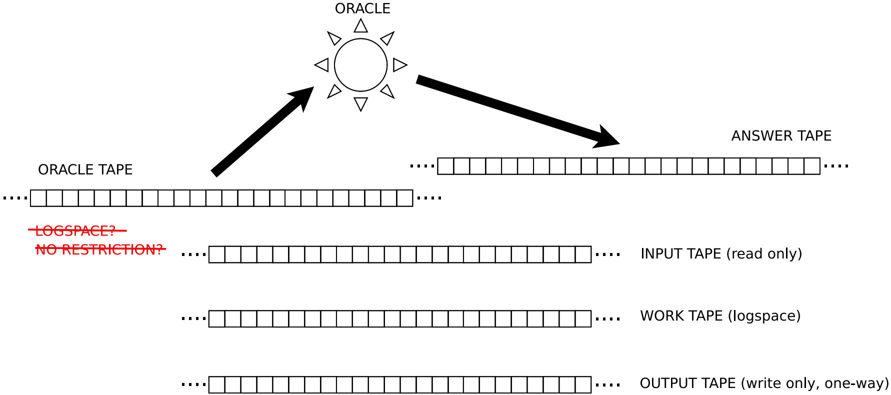
NO RESTRICTION?

INPUT TAPE (read only)

WORK TAPE (logspace)

OUTPUT TAPE (write only, one-way)

Small Complexity Classes



ORACLE

ANSWER TAPE

ORACLE TAPE

~~LOGSPACE?~~

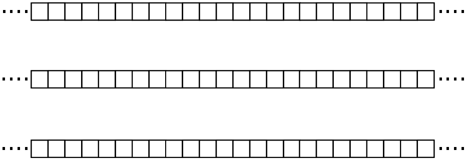
~~NO RESTRICTION?~~

INPUT TAPE (read only)

WORK TAPE (logspace)

OUTPUT TAPE (write only, one-way)

The stack model (Kawamura and Ota)

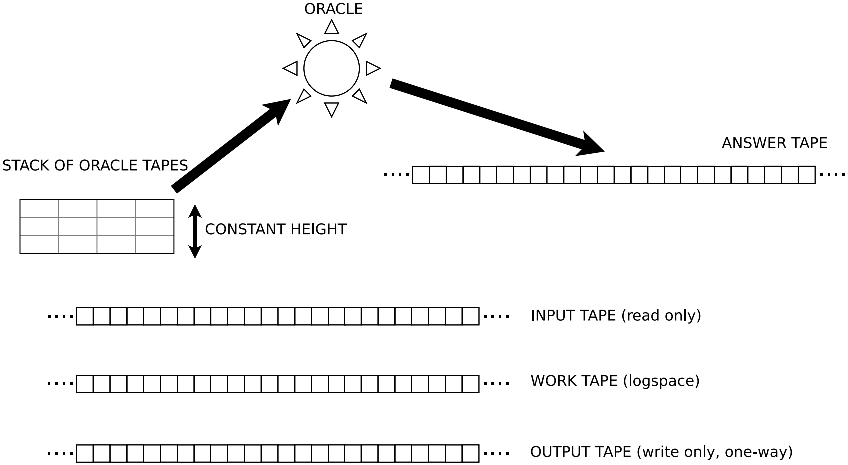


INPUT TAPE (read only)

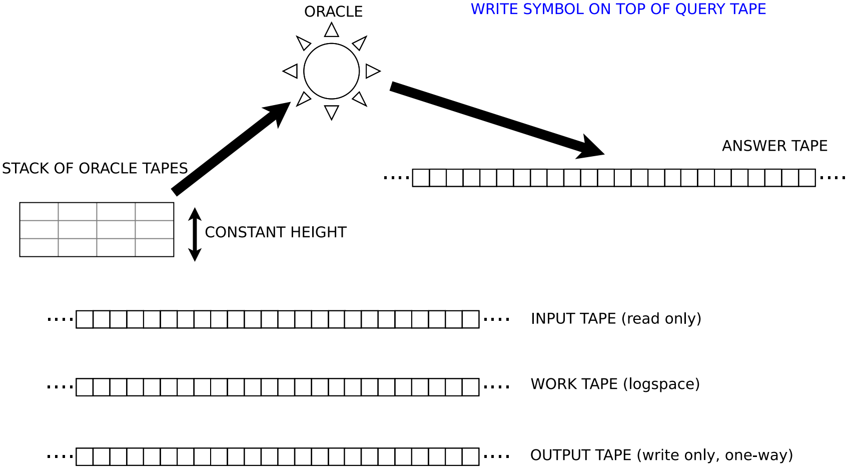
WORK TAPE (logspace)

OUTPUT TAPE (write only, one-way)

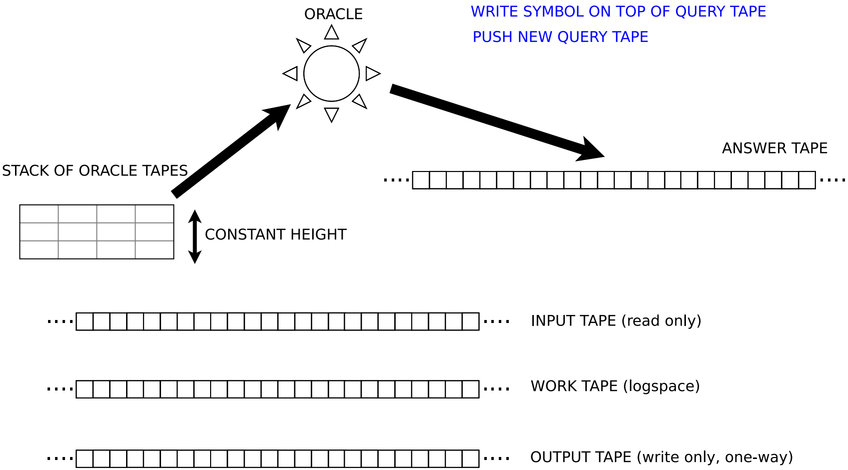
The stack model (Kawamura and Ota)



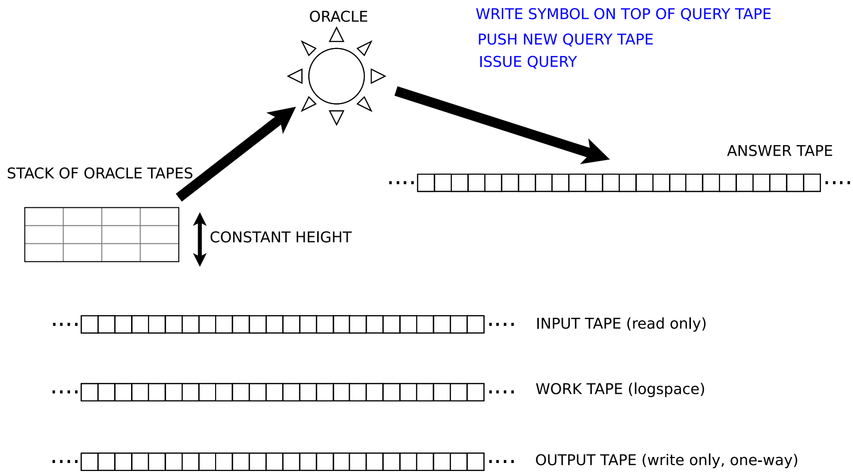
The stack model (Kawamura and Ota)



The stack model (Kawamura and Ota)



The stack model (Kawamura and Ota)



WRITE SYMBOL ON TOP OF QUERY TAPE

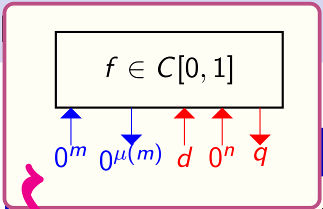
PUSH NEW QUERY TAPE

ISSUE QUERY

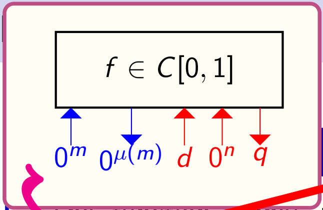
Example

Function Application

The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}, (f,x) \mapsto f(x)$ is $([\delta_{\square}, \rho_{\mathbb{R}}^{|[0,1]}], \rho_{\mathbb{R}})-$FL computable.

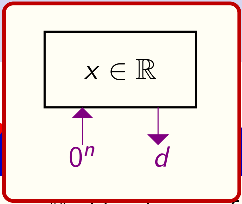


The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{|[0,1]}], \rho_{\mathbb{R}})$−FL computable.



$f \in C[0,1]$

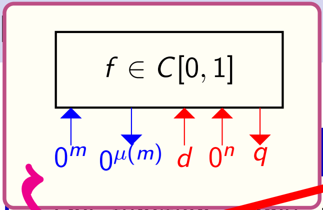
0^m $0^{\mu(m)}$ d 0^n q

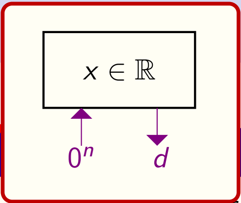


$x \in \mathbb{R}$

0^n d

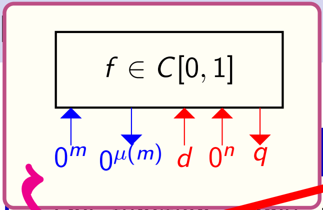
The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})-$FL computable.

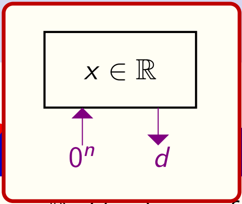




The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_\mathbb{R}^{|[0,1]}], \rho_\mathbb{R})$−FL computable.

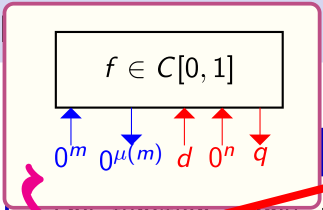
Input:	0^n
Stack:	-
Oracle Output:	ε
Work:	ε
Output:	ε

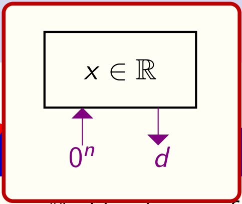




The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})$−FL computable.

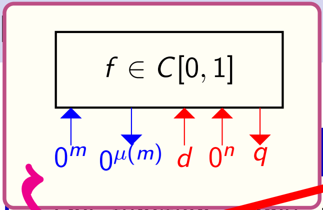
```
Input:          0^n
Stack:          ε;ε;ε
Oracle Output:  ε
Work:           ε
Output:         ε
```

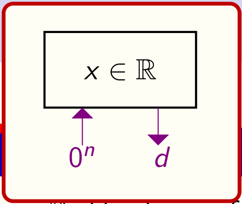




The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})$−FL computable.

Input:	0^n
Stack:	$0^{n+1};\varepsilon;\varepsilon$
Oracle Output:	ε
Work:	ε
Output:	ε

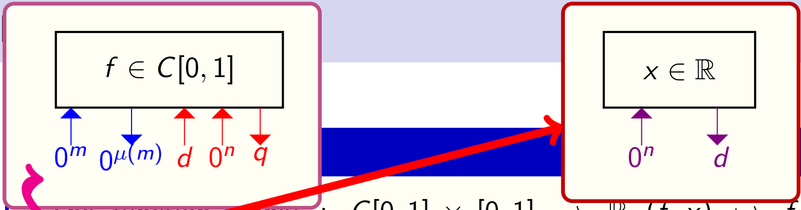




The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})$−FL computable.

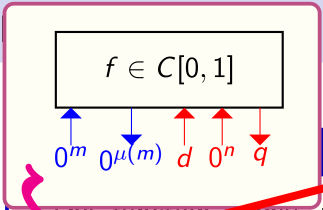
`Input:`	0^n
`Stack:`	ε;ε
`Oracle Output:`	0^m
`Work:`	ε
`Output:`	ε

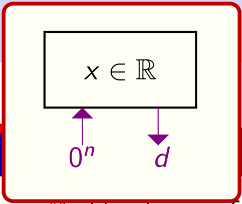
$$|x-y| \le 2^{-m} \Rightarrow |f(x)-f(y)| \le 2^{-(n+1)}$$



The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_\mathbb{R}^{[0,1]}], \rho_\mathbb{R})$−FL computable.

Input:	0^n
Stack:	0^m;ε
Oracle Output:	0^m
Work:	ε
Output:	ε

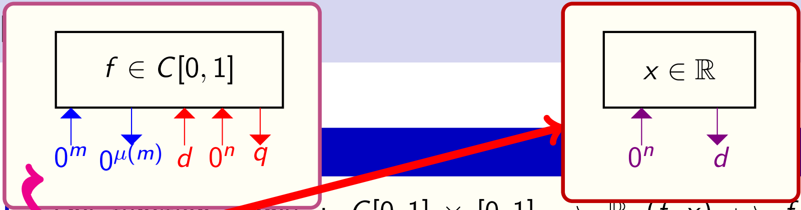




The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}, (f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})$−FL computable.

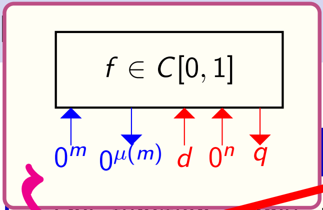
`Input:`	0^n
`Stack:`	ε
`Oracle Output:`	d
`Work:`	ε
`Output:`	ε

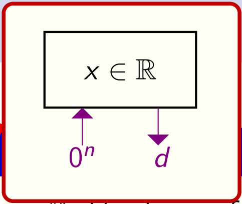
$|x - d| \leq 2^{-m}$



The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\Box, \rho_{\mathbb{R}}^{[0,1]}], \rho_{\mathbb{R}})$−FL computable.

`Input:`	0^n
`Stack:`	$d, 0^{n+1}$
`Oracle Output:`	d
`Work:`	ε
`Output:`	ε

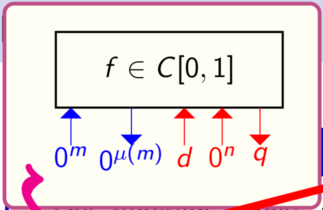


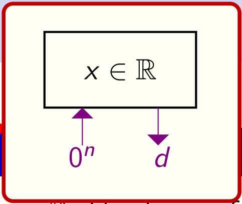


The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}, (f,x) \mapsto f(x)$ is $([\delta_\square, \rho_\mathbb{R}^{[0,1]}], \rho_\mathbb{R})$−FL computable.

`Input:`	0^n
`Stack:`	-
`Oracle Output:`	q
`Work:`	ε
`Output:`	ε

$$|q - f(d)| \leq 2^{-(n+1)}$$





The function Apply : $C[0,1] \times [0,1] \to \mathbb{R}$, $(f,x) \mapsto f(x)$ is $([\delta_\square, \rho_{\mathbb{R}}^{|[0,1]}], \rho_{\mathbb{R}})$−FL computable.

`Input:`	0^n
`Stack:`	-
`Oracle Output:`	q
`Work:`	ε
`Output:`	q

$|q - f(x)| \leq 2^{-n}$

Example

Function Application

The function Apply $: C[0,1] \times [0,1] \to \mathbb{R}, (f,x) \mapsto f(x)$ is $([\delta_\square, \rho_\mathbb{R}^{|[0,1]}], \rho_\mathbb{R})-$FL computable.

Similarly, Apply^c $: C[0,1] \times [0,1] \to [0,1]\,(f,x) \mapsto f^c(x)$ is $([\delta_\square, \rho_\mathbb{R}^{|[0,1]}], \rho_\mathbb{R})-$FL computable for constant $c \in \mathbb{N}$ using a stack of size $2c+1$.

Some Results from Real Complexity Theory

Fact

For general polynomial time computable functions, many important operators have been shown to be computationally hard.

For example

- Polynomial time computable functions may have non computable derivatives. (Ko 1983)
- Parametric maximization is NP-hard. (Ko/Friedman (1982))
- Integration is #P-hard. (Friedman (1984))

Analytic Function

An analytic function is a function locally given by a complex power series.

Definition (Analytic Function)

$f : D \to \mathbb{C}$, $D \subseteq \mathbb{C}$ is analytic if for any $x_0 \in D$ the Taylor-series

$$T(x) := \sum_{n=0}^{\infty} a_n (x - x_0)^n$$

converges to $f(x)$ for x in a neighborhood of x_0.

Analytic Function

An analytic f es.

Computable Sequence

$(a_m)_{m \in \mathbb{N}}$ is computable if there is a machine that on input $1^n, 1^m$ outputs a rational approximation $d_{n,m}$ with $|d_{n,m} - a_m| \leq 2^{-n}$

Definition (A

$f : D \to \mathbb{C}$, D

$$T(x) := \sum_{n=0}^{\infty} a_n (x - x_0)^n$$

converges to $f(x)$ for x in a neighborhood of x_0.

Analytic Function

Computable Sequence

$(a_m)_{m\in\mathbb{N}}$ is computable if there is a machine that on input $1^n, 1^m$ outputs a rational approximation $d_{n,m}$ with $|d_{n,m} - a_m| \leq 2^{-n}$

An analytic f es.

Definition (A

$f : D \to \mathbb{C}$, D

$$T(x) := \sum_{n=0}^{\infty} a_n(x - x_0)^n$$

converges to $f(x)$ for x in a neighborhood of x_0.

Theorem (Pour-El, Richards, Ko, Friedman, Müller (1987/1989))

f is (polytime) computable iff $(a_m)_{m\in\mathbb{N}}$ is.

From that polynomial time computability of the derivative and the anti-derivative of a function follows immediately.

Representation for Analytic Functions

How to represent analytic functions?

Representation for Analytic Functions

How to represent analytic functions?

Theorem (Müller (1995))

The evaluation operator $((a_m)_{m\in\mathbb{N}}, x) \to f(x)$ *is not computable.*

Representation for Analytic Functions

How to represent analytic functions?

Theorem (Müller (1995))

The evaluation operator $((a_m)_{m\in\mathbb{N}}, x) \to f(x)$ is not computable.

Lemma

Let $f : \overline{B}(0,1) \to \mathbb{R}$ be analytic and $(a_n)_{n\in\mathbb{N}}$ its power series around 0. Then there exist $k, A \in \mathbb{N}$ such that

1. $\sqrt[k]{2}$ *is a lower bound on the radius of convergence*
2. $|a_n| \leq A \cdot 2^{-\frac{n}{k}}$

How to represent analytic functions?

Representation 1

A function $\varphi \in \Sigma^{**}$ is a name for a power series $(a_k)_{k\in\mathbb{N}}$ iff it is a concatenation of the following

1. *An integer A encoded in binary*
2. *An integer k encoded in unary*
3. *A name for a sequence $(a_k)_{k\in\mathbb{N}}$*

Such that $|a_n| \leq A \cdot 2^{-\frac{n}{k}}$ for all $n \in \mathbb{N}$.

How to represent analytic functions?

Representation 2

A (length-monotone) function $\varphi : \Sigma^* \to \Sigma^*$ *is a name for an analytic function* $f : \bar{B}(0,1) \to \mathbb{R}$ *iff it is a concatenation of the following*

1. *An integer* A *encoded in binary,*
2. *An integer* k *encoded in unary,*
3. *A name for the function* f

Such that f *extends analytically to* $B(0, \sqrt[k]{2})$ *and* $|f(z)| \leq A$ *for all* $z \in B(0, \sqrt[k]{2})$

Analytic Functions and Computational Complexity

Theorem (Kawamura, Rösnick, Müller, Ziegler (2013))

With the previous two representations the following operations can be performed in polynomial time

1. *evaluation*
2. *addition and multiplication*
3. *differentiation and anti-differentiation*
4. *parametric maximization*

Analytic Functions and Computational Complexity

Theorem (Kawamura, Rösnick, Müller, Ziegler (2013))

With the previous two representations the following operations can be performed in polynomial time

1. *evaluation*
2. *addition and multiplication*
3. *differentiation and anti-differentiation*
4. *parametric maximization*

Further, when identifying an analytic function with its power series, the operators that compute one representation from the other are polynomial-time computable.

Complexity of Ordinary Differential Equations

Theorem (Kawamura, 2010)

Consider the IVP

$$y'(t) = f(t, y(t)) \quad ; \quad y(0) = 0.$$

There exists functions $f : [0,1] \times [-1,1] \to \mathbb{R}$ *and* $y : [0,1] \to [-1,1]$ *as above such that*

1. f *is Lipschitz-continuous and polynomial time computable*
2. y *is PSPACE-hard.*

Complexity of Ordinary Differential Equations

Theorem (Kawamura, 2010)

Consider the IVP

$$y'(t) = f(t, y(t)) \quad ; \quad y(0) = 0.$$

There exists functions $f : [0,1] \times [-1,1] \to \mathbb{R}$ *and* $y : [0,1] \to [-1,1]$ *as above such that*

1. f *is Lipschitz-continuous and polynomial time computable*
2. y *is PSPACE-hard.*

For analytic right-hand side y will again be analytic and polynomial time computable.

Analytic Functions and Small Complexity Classes

Consider functions complex analytic on the closed unit disc.

Representation 1

Integers A, k and the series sequence $(a_k)_{k \in \mathbb{N}}$.
$|a_n| \leq A \cdot 2^{-\frac{n}{k}}$ *for all* $n \in \mathbb{N}$.

Representation 2

Integers A, k, and name for function f.
f *extends analytically to* $B(0, \sqrt[k]{2})$ *and* $|f(z)| \leq A$ *for all* $z \in B(0, \sqrt[k]{2})$

Those two representations are logspace equivalent.

Representation 1 $\Rightarrow$ Representation 2

Task

Given A, k s.t. $|a_n| \leq A \cdot 2^{-\frac{n}{k}}$ for all $n \in \mathbb{N}$.
Find A', k' s.t. for all $z \in B(0, \sqrt[k']{2})$ $|f(z)| \leq A'$

Representation 1 $\Rightarrow$ Representation 2

Task

Given A, k s.t. $|a_n| \leq A \cdot 2^{-\frac{n}{k}}$ for all $n \in \mathbb{N}$.
Find A', k' s.t. for all $z \in B(0, \sqrt[k']{2})$ $|f(z)| \leq A'$

Solution

Let $k' = 2k$ and $A' = 4kA$

Representation 1 $\Rightarrow$ Representation 2

Task

Given A, k s.t. $|a_n| \leq A \cdot 2^{-\frac{n}{k}}$ for all $n \in \mathbb{N}$.
Find A', k' s.t. for all $z \in B(0, \sqrt[k']{2})$ $|f(z)| \leq A'$

Solution

Let $k' = 2k$ and $A' = 4kA$

Proof

For $z \in B(0, \sqrt[k']{2}) :\ f(z) = \sum_{n=0}^{\infty} a_n z^n \leq A \sum_{n=0}^{\infty} 2^{-\frac{n}{k}} \cdot 2^{\frac{n}{2k}} \leq 4Ak$

Representation 1 ⇒ Representation 2

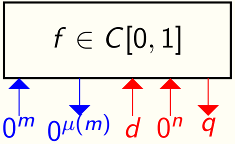
Task

Compute a function name for f.

Representation 1 ⇒ Representation

Task

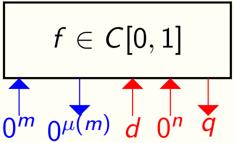
Compute a function name for f.



$f \in C[0,1]$

0^m $0^{\mu(m)}$ d 0^n q

Representation 1 ⇒ Representation



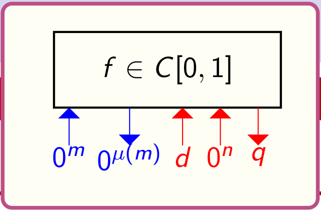
Task

Compute a function name for f.

Solution

$m \mapsto m + \log_2(A) + 2\log_2(k) + 5$ is a modulus of continuity for the function.

Representation 1 ⇒ Representation



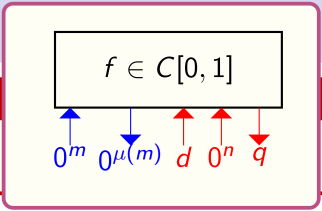
Task

Compute a function name for f.

Solution

$m \mapsto m + \log_2(A) + 2\log_2(k) + 5$ is a modulus of continuity for the function.

Representation 1 ⇒ Representation



$f \in C[0,1]$

0^m $0^{\mu(m)}$ d 0^n q

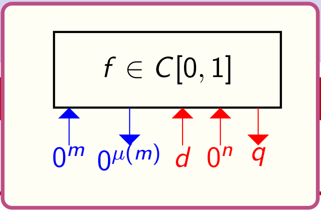
Task

Compute a function name for f.

Solution

We need to evaluate $N \approx n \cdot k + \log(A)$ terms of the sum.
$\sum_{j=0}^{poly(n)} a_j x^j$ is logspace computable.

Representation 1 ⇒ Representation



Task

Compute a function name for f.

Solution

We need to evaluate $N \approx n \cdot k + \log(A)$ terms of the sum.
$\sum_{j=0}^{poly(n)} a_j x^j$ is logspace computable.

Proof

The following is logspace computable

1. Addition and Multiplication of polynomially many n-bit integers
2. x^m with precision polynomial in n for polynomial length m
3. Composition of constantly many logspace computable functions

Representation 2 $\Rightarrow$ Representation 1

Task

Given A, k s.t. for all $z \in B(0, \sqrt[k']{2})$ $|f(z)| \leq A$
Compute A', k' such that $|a_m| \leq A \cdot 2^{-\frac{m}{k}}$ for all $m \in \mathbb{N}$.

Solution

We can just set $A' = A$ and $k' = k$.

Representation 2 $\Rightarrow$ Representation 1

Task

Given A, k s.t. for all $z \in B(0, \sqrt[k']{2})$ $|f(z)| \leq A$
Compute A', k' such that $|a_m| \leq A \cdot 2^{-\frac{m}{k}}$ for all $m \in \mathbb{N}$.

Solution

We can just set $A' = A$ and $k' = k$.

Proof

By Cauchy's integral formula $|a_m| = \frac{f^{(m)}(0)}{m!} \leq A \cdot 2^{-\frac{n}{k}}$ for all $n \in \mathbb{N}$.

Representation 2 $\Rightarrow$ Representation 1

Task

Compute the coefficients for the power series around 0.

Representation 2 $\Rightarrow$ Representation 1

Task

Compute the coefficients for the power series around 0.

Solution

- Approximate function by a polynomial
- Differentiate this polynomial and evaluate at 0
- Divide by factorial

Representation 2 ⇒ Representation 1

Task

Compute the coefficients for the power series around 0.

Solution

- Approximate function by a polynomial
- Differentiate this polynomial and evaluate at 0
- Divide by factorial

Proof

- Computing factorials and binomial coefficients of polynomial size is logspace computable.
- To get the coefficient a_m the function has to be evaluated at $2m+1$ equidistant points with polynomial precision (see e.g.Müller).

Further Operations

Logspace computable operations

Similarly, the following operations on analytic functions are computable in logartihmic-space

1. Addition, Subtraction, Multiplication of two analytic functions

Further Operations

Logspace computable operations

Similarly, the following operations on analytic functions are computable in logartihmic-space

1. Addition, Subtraction, Multiplication of two analytic functions
2. Computing the d-fold derivative

Further Operations

Logspace computable operations

Similarly, the following operations on analytic functions are computable in logartihmic-space

1. Addition, Subtraction, Multiplication of two analytic functions
2. Computing the d-fold derivative
3. Computing the d-fold anti-derivative

Multidimensional Analytic Functions

Multidimensional Power Series

$$\begin{aligned}\sum_{i\in\mathbb{N}}\sum_{j\in\mathbb{N}} a_{i,j}x_1^i x_2^j &= \sum_{i\in\mathbb{N}} b_i x_1^i \\ \text{with } b_i &:= \sum_{j\in\mathbb{N}} a_{i,j}x_2^j\end{aligned}$$

Computing $b_i \rightarrow$ evaluating an analytic function.

Multidimensional Analytic Functions

Multidimensional Power Series

$$\begin{aligned} \sum_{i\in\mathbb{N}}\sum_{j\in\mathbb{N}} a_{i,j}x_1^i x_2^j &= \sum_{i\in\mathbb{N}} b_i x_1^i \\ \text{with } b_i &:= \sum_{j\in\mathbb{N}} a_{i,j}x_2^j \end{aligned}$$

Computing b_i → evaluating an analytic function.

$|a_{i,j}| \leq A \cdot 2^{-\frac{i+j}{k}}$ for all $n \in \mathbb{N}$.

$|b_i| \leq \sum_{j\in\mathbb{N}} |a_{i,j}|\,|x_2|^j \leq A2^{-\frac{i}{k}} \sum_{j\in\mathbb{N}} 2^{-\frac{j}{k}} = A2^{-\frac{j}{k}} \frac{\sqrt[k]{2}}{\sqrt[k]{2}-1} \leq \mathbf{(2Ak)}2^{-\frac{i}{k}}$

Multidimensional analytic functions

Representation

A function $\varphi \in \Sigma^{**}$ is a name for a d dimensional power series $(a_{n_1,\dots,n_d})_{n_1,\dots,n_d \in \mathbb{N}}$ iff it is a concatenation of the following

1. An integer A encoded in binary
2. An integer k encoded in unary
3. A name for a sequence $(a_k)_{k \in \mathbb{N}}$

Such that $|a_{n_1,\dots,n_d}| \leq A \cdot 2^{-\frac{n_1+\dots+n_d}{k}}$ for all $n \in \mathbb{N}$.

Multidimensional analytic functions

Representation

A function $\varphi \in \Sigma^{**}$ is a name for a d dimensional power series $(a_{n_1,\dots,n_d})_{n_1,\dots,n_d \in \mathbb{N}}$ iff it is a concatenation of the following

1. An integer A encoded in binary
2. An integer k encoded in unary
3. A name for a sequence $(a_k)_{k \in \mathbb{N}}$

Such that $|a_{n_1,\dots,n_d}| \leq A \cdot 2^{-\frac{n_1+\cdots+n_d}{k}}$ for all $n \in \mathbb{N}$.

Note

Complexity of evaluation (and most other operations) is **exponential** in the dimension!

Open Problem: P-completeness

Kawamura and Ota also define the notions of reductions and completeness.

Open Problem: P-completeness

Kawamura and Ota also define the notions of reductions and completeness. For example they give the following uniform version of a theorem by Ko

Theorem

For the set M of bijective functions in $C[0,1]$, define the $\delta_{\square INV}$ representation by adding a modulus of continuity for the inverse function to the $\delta_{\square}$ representation.
The function $Inv : M \to C[0,1]$, $Inv(f) \mapsto f^{-1}$ is $(\delta_{\square INV}, \delta_{\square})$-FP-complete.

Open Problem: P-completeness

Kawamura and Ota also define the notions of reductions and completeness. For example they give the following uniform version of a theorem by Ko

Theorem

For the set M of bijective functions in $C[0,1]$, define the $\delta_{\Box INV}$ representation by adding a modulus of continuity for the inverse function to the $\delta_{\Box}$ representation.
The function $Inv : M \to C[0,1]$, $Inv(f) \mapsto f^{-1}$ is $(\delta_{\Box INV}, \delta_{\Box})$-FP-complete.

Which operations on analytic functions are P-hard?

- Initial Value Problem?
- Maximization?

Open Problem: P-completeness

Initial Value Problem

$$\dot{y}(t) = f(y(t)) \quad ; \quad y(0) = 0.$$

Open Problem: P-completeness

Initial Value Problem

$$\dot{y}(t) = f(y(t)) \quad ; \quad y(0) = 0.$$

Power series

$$\begin{aligned} f^{[0]}(y) &= y \\ f^{[k+1]}(y) &= \frac{1}{k+1} f(y) \frac{d}{dy} f^{[k]}(y) \end{aligned}$$

Then

$$a_k = \frac{1}{k!} y^{(k)}(0) = f^{[k]}(0)$$

This can be used to get a recurrence relation that makes it possible to compute the coefficients in polynomial time.

Can they also be computed without such a recurrence?

Conclusion

- Presented Kawamura and Ota's model for logspace computability in analysis.
- In this model many operations on analytic functions are logspace computable, when considering representations that have previously been considered for polynomial time computability.
- Open Problems: Parametrized Maximization, Ordinary Differential Equations
- Connection to parallelization in exact real arithmetic
- Implementations