Scommettere aiuta:
Certezze infinite ed errori consapevoli

Guido Gherardi
Joint work with Vasco Brattka and Rupert Hölzl

Fakultät für Informatik
Universität der Bundeswehr München

Torino, 17.06.2015

Π_2-Statements

$$(\forall x \in X)(\exists y \in Y)R(x, y)$$

The Computable Analysis Model

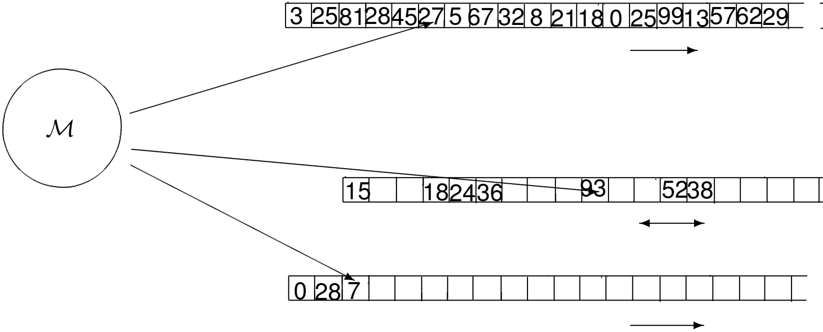


Figure : A Turing machine working with infinite sequences.

Definition (Representations)

A representation of a set X is a surjective function $\rho_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$.

The pair (X, ρ_X) is called a represented space.

Usually, admissible representations (e.g., Cauchy representartions) are used.

$x \in X$ is ρ_X-computable if it has some computable ρ_X-name $p \in \mathbb{N}^{\mathbb{N}}$.

Definition (Representations)

A representation of a set X is a surjective function $\rho_X :\subseteq \mathbb{N}^{\mathbb{N}} \to X$.

The pair (X, ρ_X) is called a represented space.

Usually, admissible representations (e.g., Cauchy representartions) are used.

$x \in X$ is ρ_X-computable if it has some computable ρ_X-name $p \in \mathbb{N}^{\mathbb{N}}$.

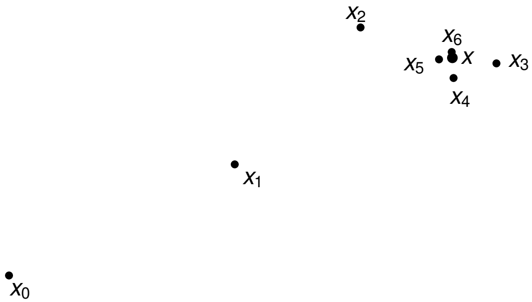


Figure : The Cauchy Representation δ_X: for a separable metric space X, a point $x \in X$ is encoded by a Cauchy sequence $x_0, x_1, x_2, ...$ of elements from a fixed dense countable set $D \subseteq X$ that *uniformly* converges to x.

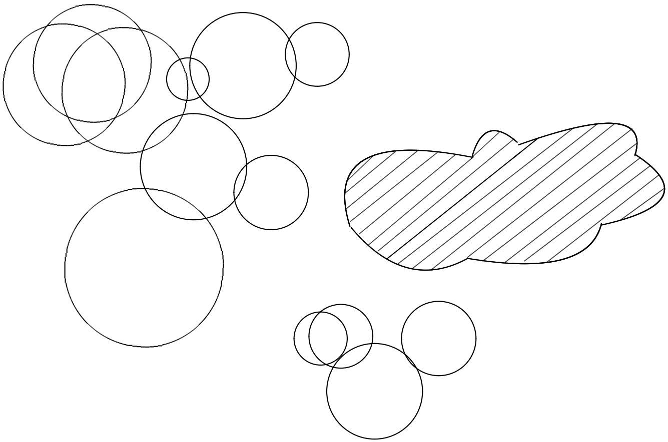


Figure : The Negative Representation ψ_X: for a separable metric space X, a closed set $A \subseteq X$ is encoded by a list of basic open balls exausting its complement.

Definition (Realizers)

Let (X, ρ_X), (Y, ρ_Y) be represented spaces and let $f :\subseteq X \rightrightarrows Y$ be a multi-valued function. A function $F :\subseteq \mathbb{N}^\mathbb{N} \to \mathbb{N}^\mathbb{N}$ is a (ρ_X, ρ_Y)-realizer of f ($F \vdash f$) if $\rho_Y \circ F(p) \in f(\rho_X(p))$ for all $p \in \mathbb{N}^\mathbb{N}$ such that $\rho_X(p) \in \mathrm{dom}(f)$.

$$\begin{array}{ccc} p \in \mathbb{N}^\mathbb{N} & \xrightarrow{F} & F(p) \in \mathbb{N}^\mathbb{N} \\ \Big\downarrow{\scriptstyle \rho_X} & & \Big\downarrow{\scriptstyle \rho_Y} \\ x \in X & \underset{f}{\Longrightarrow} & y \in f(x) \subseteq Y \end{array}$$

f is said to be (ρ_X, ρ_Y)-computable if it has a computable (ρ_X, ρ_Y)-realizer F.

Definition (Realizers)

Let (X, ρ_X), (Y, ρ_Y) be represented spaces and let $f :\subseteq X \rightrightarrows Y$ be a multi-valued function. A function $F :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ is a (ρ_X, ρ_Y)-realizer of f ($F \vdash f$) if $\rho_Y \circ F(p) \in f(\rho_X(p))$ for all $p \in \mathbb{N}^{\mathbb{N}}$ such that $\rho_X(p) \in \mathrm{dom}(f)$.

$$\begin{array}{ccc} p \in \mathbb{N}^{\mathbb{N}} & \xrightarrow{F} & F(p) \in \mathbb{N}^{\mathbb{N}} \\ \downarrow{\scriptstyle \rho_X} & & \downarrow{\scriptstyle \rho_Y} \\ x \in X & \underset{f}{\rightrightarrows} & y \in f(x) \subseteq Y \end{array}$$

f is said to be (ρ_X, ρ_Y)-computable if it has a computable (ρ_X, ρ_Y)-realizer F.

Paradigm extensions: limit computability

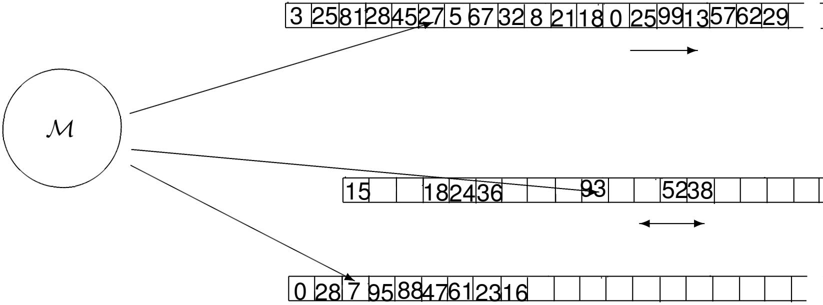
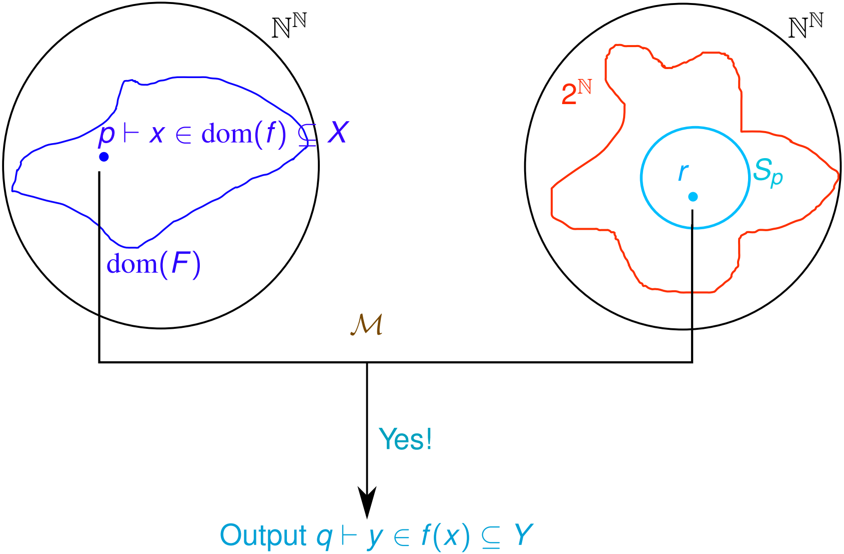


Figure : A limit Turing machine.

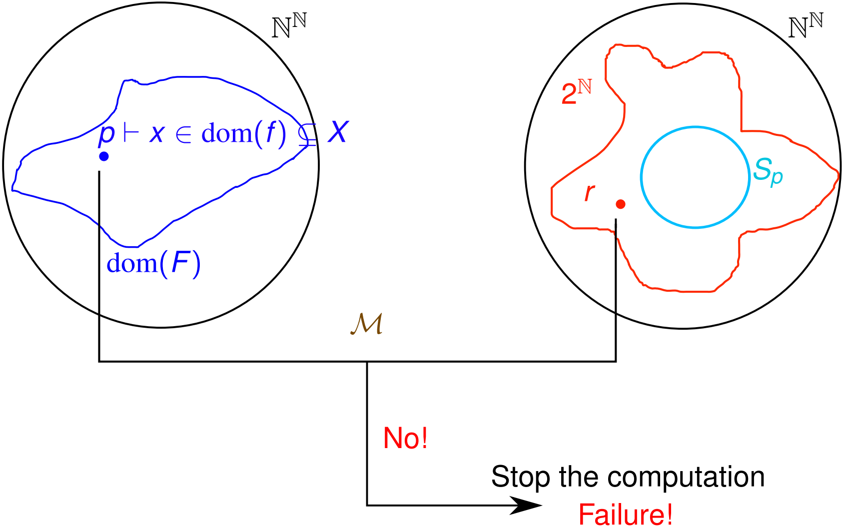
Paradigm extensions: non deterministic computations

A (multi-valued) function $f :\subseteq X \rightrightarrows Y$ is said to be
non deterministic computable if the following conditions hold:

Succes oracles

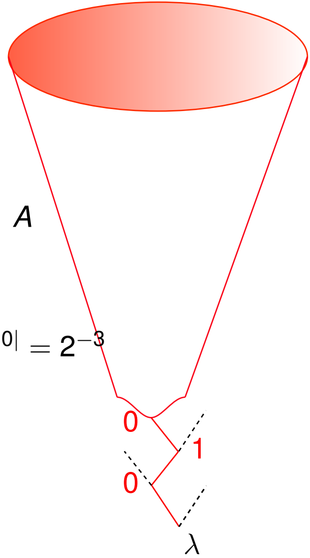


Failure recognition mechanism



Las Vegas computability

The closed set S_p has moreover positive measure $\mu(S_p) > 0$:



$$\mu(A) = \mu(0102^{\mathbb{N}}) = 2^{-|010|} = 2^{-3}$$

Examples of theorems that are:

- computable: Urysohn Extension Lemma, Urysohn-Tietze Extension Lemma, Heine-Borel Theorem, Weierstrass Approximation Theorem, Baire Category Theorem,...
- finitely mind changes complete:, Banach Inverse Mapping Theorem, Open Mapping Theorem, Closed Graph Theorem, Uniform Boundedness Theorem, (contrapositive of) Baire Category Theorem...
- non deterministic complete: Hahn-Banach Extension Theorem, Brouwer Fixed Point Theorem,...
- limit complete: Monotone Convergence Theorem,...

Examples of theorems that are:

- computable: Urysohn Extension Lemma, Urysohn-Tietze Extension Lemma, Heine-Borel Theorem, Weierstrass Approximation Theorem, Baire Category Theorem,...
- finitely mind changes complete:, Banach Inverse Mapping Theorem, Open Mapping Theorem, Closed Graph Theorem, Uniform Boundedness Theorem, (contrapositive of) Baire Category Theorem...
- non deterministic complete: Hahn-Banach Extension Theorem, Brouwer Fixed Point Theorem,...
- limit complete: Monotone Convergence Theorem,...

Examples of theorems that are:

- computable: Urysohn Extension Lemma, Urysohn-Tietze Extension Lemma, Heine-Borel Theorem, Weierstrass Approximation Theorem, Baire Category Theorem,...
- finitely mind changes complete:, Banach Inverse Mapping Theorem, Open Mapping Theorem, Closed Graph Theorem, Uniform Boundedness Theorem, (contrapositive of) Baire Category Theorem...
- non deterministic complete: Hahn-Banach Extension Theorem, Brouwer Fixed Point Theorem,...
- limit complete: Monotone Convergence Theorem,...

Examples of theorems that are:

- computable: Urysohn Extension Lemma, Urysohn-Tietze Extension Lemma, Heine-Borel Theorem, Weierstrass Approximation Theorem, Baire Category Theorem,...
- finitely mind changes complete:, Banach Inverse Mapping Theorem, Open Mapping Theorem, Closed Graph Theorem, Uniform Boundedness Theorem, (contrapositive of) Baire Category Theorem...
- non deterministic complete: Hahn-Banach Extension Theorem, Brouwer Fixed Point Theorem,...
- limit complete: Monotone Convergence Theorem,...

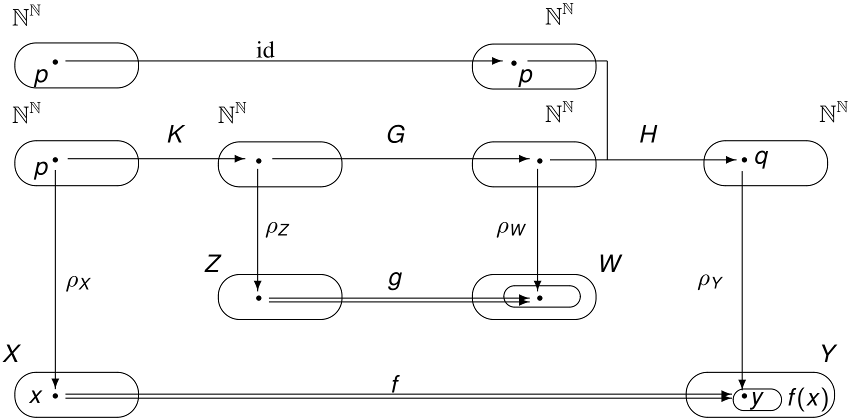
Examples of theorems that are:

- computable: Urysohn Extension Lemma, Urysohn-Tietze Extension Lemma, Heine-Borel Theorem, Weierstrass Approximation Theorem, Baire Category Theorem,...
- finitely mind changes complete:, Banach Inverse Mapping Theorem, Open Mapping Theorem, Closed Graph Theorem, Uniform Boundedness Theorem, (contrapositive of) Baire Category Theorem...
- non deterministic complete: Hahn-Banach Extension Theorem, Brouwer Fixed Point Theorem,...
- limit complete: Monotone Convergence Theorem,...

Characterization of Las Vegas computable functions

f is Weihrauch-reducible to g ($f \leq_W g$) if there are computable $K :\subseteq \mathbb{N}^{\mathbb{N}} \times \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$, $H :\subseteq \mathbb{N}^{\mathbb{N}} \to \mathbb{N}^{\mathbb{N}}$ such that

$$H(id, GK) \vdash f, \text{ for every } G \vdash g.$$



$<_W$, $\equiv_W$, $|_W$ are defined in the obvious way.

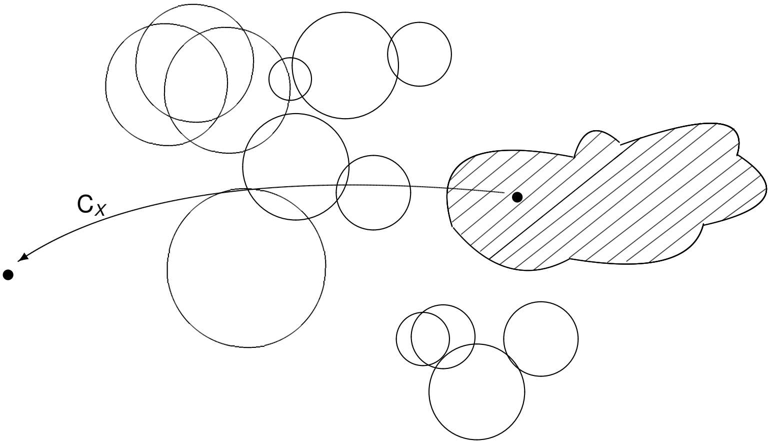
Via $\equiv_W$ a lattice of equivalence classes is obtained, called the Weihrauch lattice.

$<_W$, $\equiv_W$, $|_W$ are defined in the obvious way.

Via $\equiv_W$ a lattice of equivalence classes is obtained, called the Weihrauch lattice.

Definition (Positive Closed Choice)

Given a computable metric space X with a Borel measure μ, $\mathsf{PC}_X :\subseteq \mathcal{A}_-(X) \rightrightarrows X, A \mapsto A$, is the positive closed choice operator, which selects points from closed sets $A \subseteq X$ of positive measure denoted by the negative representation ψ_-^X.



Theorem

Let X and Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas computable,*
- $f \leq_{\mathrm{W}} \mathrm{PC}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathrm{PC}_{[0,1]}$.

Theorem

Let X and Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas computable,*
- $f \leq_{\mathrm{W}} \mathrm{PC}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathrm{PC}_{[0,1]}$.

Theorem

Let X and Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas computable,*
- $f \leq_{\mathrm{W}} \mathrm{PC}_{2^{\mathbb{N}}} \equiv_{\mathrm{W}} \mathrm{PC}_{[0,1]}$.

Definition (WWKL)

Let $T^\infty \subseteq 2^{\mathbb{N}}$ be the set of (charachteristic functions of) infinite binary trees. We define then:

$\mathrm{WWKL} :\subseteq T^\infty \rightrightarrows 2^{\mathbb{N}}$,
$T \mapsto [T] := \{p \in 2^{\mathbb{N}} | \, p \text{ is an infinite path in } T \text{ with } \mu([T]) > 0\}$.

Theorem

Let X and Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas computable,*
- *$f \leq_{\mathrm{W}} \mathrm{WWKL}$.*

Definition (WWKL)

Let $T^{\infty} \subseteq 2^{\mathbb{N}}$ be the set of (charachteristic functions of) infinite binary trees. We define then:

$$\mathrm{WWKL} :\subseteq T^{\infty} \rightrightarrows 2^{\mathbb{N}},$$
$$T \mapsto [T] := \{p \in 2^{\mathbb{N}} | \, p \textit{ is an infinite path in } T \textit{ with } \mu([T]) > 0\}.$$

Theorem

Let X and Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas computable,*
- *$f \leq_{\mathrm{W}} \mathrm{WWKL}$.*

Basic properties of Las Vegas Computable Functions

Theorem (Closure under composition)

The class of Las Vegas computable multi-valued function is closed under composition.

Hence: Las Vegas Computable multi-valued functions constitute a natural computational class.

Basic properties of Las Vegas Computable Functions

Theorem (Closure under composition)

The class of Las Vegas computable multi-valued function is closed under composition.

Hence: Las Vegas Computable multi-valued functions constitute a natural computational class.

Do interesting Las Vegas computable functions exist?

VITALI'S THEOREM

Let $A \subseteq [0,1]$ be Lebesgue measurable and let $\mathcal{I}$ be a sequence of intervals. If $\mathcal{I}$ is a Vitali cover of A, then there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates A.

Three classically equivalent versions of the Vitali Covering Theorem (for the special case of $A = [0,1]$):

1. For every Vitali cover $\mathcal{I}$ of $[0,1]$ there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates $[0,1]$.
2. For every saturated sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0,1]$, there exists a point $x \in [0,1]$ that is not covered by $\mathcal{I}$.
3. For every sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0,1]$, there exists a point $x \in [0,1]$ that is not captured by $\mathcal{I}$.

Do interesting Las Vegas computable functions exist?

VITALI'S THEOREM

Let $A \subseteq [0, 1]$ be Lebesgue measurable and let $\mathcal{I}$ be a sequence of intervals. If $\mathcal{I}$ is a Vitali cover of A, then there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates A.

Three classically equivalent versions of the Vitali Covering Theorem (for the special case of $A = [0, 1]$):

1. For every Vitali cover $\mathcal{I}$ of $[0, 1]$ there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates $[0, 1]$.
2. For every saturated sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0, 1]$, there exists a point $x \in [0, 1]$ that is not covered by $\mathcal{I}$.
3. For every sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0, 1]$, there exists a point $x \in [0, 1]$ that is not captured by $\mathcal{I}$.

Do interesting Las Vegas computable functions exist?

VITALI'S THEOREM

Let $A \subseteq [0,1]$ be Lebesgue measurable and let $\mathcal{I}$ be a sequence of intervals. If $\mathcal{I}$ is a Vitali cover of A, then there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates A.

Three classically equivalent versions of the Vitali Covering Theorem (for the special case of $A = [0,1]$):

1. For every Vitali cover $\mathcal{I}$ of $[0,1]$ there exists a subsequence $\mathcal{J}$ of $\mathcal{I}$ that eliminates $[0,1]$.
2. For every saturated sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0,1]$, there exists a point $x \in [0,1]$ that is not covered by $\mathcal{I}$.
3. For every sequence $\mathcal{I}$ that does not admit a subsequence which eliminates $[0,1]$, there exists a point $x \in [0,1]$ that is not captured by $\mathcal{I}$.

The corresponding multi-valued functions:

Definition (Vitali Covering Theorem)

We define the following:

1. $\mathsf{VCT}_0 :\subseteq \mathrm{Int} \rightrightarrows \mathrm{Int}$ with

   $$\mathsf{VCT}_0(\mathcal{I}) := \{\mathcal{J} : \mathcal{J} \text{ is a subsequence of } \mathcal{I} \text{ that eliminates } [0,1]\}$$

   for $\mathcal{I}$ a Vitali cover of $[0,1]$.
2. $\mathsf{VCT}_1 :\subseteq \mathrm{Int} \rightrightarrows [0,1]$ with

   $$\mathsf{VCT}_1(\mathcal{I}) := [0,1] \setminus \bigcup \mathcal{I}$$

   for $\mathcal{I}$ saturated with no subsequence eliminating $[0,1]$.
3. $\mathsf{VCT}_2 :\subseteq \mathrm{Int} \rightrightarrows [0,1]$ with

   $$\mathsf{VCT}_2(\mathcal{I}) := \{x \in [0,1] : x \text{ is not captured by } \mathcal{I}\}$$

   for $\mathcal{I}$ with no subsequence eliminating $[0,1]$.

Theorem

VCT_0 *is computable.*

Theorem

VCT_1 *is Las Vegas complete.*

Las Vegas Computable functions with finitely many mind changes

If we allow the Las Vegas machines to perform finitely many corrections on the output tape, then we obtain the class of Las Vegas Computable functions with finitely many mind changes.

Theorem

The class of Las Vegas computable functions with finitely many mind changes is closed under composition.

Theorem

Let X, Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas comp. with finitely many mind changes,*
- *$f \leq_{\mathrm{W}} \mathrm{PC}_{\mathbb{R}}$.*

Proposition

$\mathrm{PC}_{\mathbb{R}} \equiv_{\mathrm{W}} \mathrm{PC}^{\Delta}_{[0,1]}$.

Las Vegas Computable functions with finitely many mind changes

If we allow the Las Vegas machines to perform finitely many corrections on the output tape, then we obtain the class of Las Vegas Computable functions with finitely many mind changes.

Theorem

The class of Las Vegas computable functions with finitely many mind changes is closed under composition.

Theorem

Let X, Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas comp. with finitely many mind changes,*
- $f \leq_W \mathrm{PC}_{\mathbb{R}}$.

Proposition

$\mathrm{PC}_{\mathbb{R}} \equiv_W \mathrm{PC}^{\Delta}_{[0,1]}$.

Las Vegas Computable functions with finitely many mind changes

If we allow the Las Vegas machines to perform finitely many corrections on the output tape, then we obtain the class of Las Vegas Computable functions with finitely many mind changes.

Theorem

The class of Las Vegas computable functions with finitely many mind changes is closed under composition.

Theorem

Let X, Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas comp. with finitely many mind changes,*
- $f \leq_{\mathrm{W}} \mathrm{PC}_{\mathbb{R}}$.

Proposition

$\mathrm{PC}_{\mathbb{R}} \equiv_{\mathrm{W}} \mathrm{PC}^{\Delta}_{[0,1]}$.

Las Vegas Computable functions with finitely many mind changes

If we allow the Las Vegas machines to perform finitely many corrections on the output tape, then we obtain the class of Las Vegas Computable functions with finitely many mind changes.

Theorem

The class of Las Vegas computable functions with finitely many mind changes is closed under composition.

Theorem

Let X, Y be represented spaces. The following are equivalent for $f :\subseteq X \rightrightarrows Y$:

- *f is Las Vegas comp. with finitely many mind changes,*
- *$f \leq_{\mathrm{W}} \mathrm{PC}_{\mathbb{R}}$.*

Proposition

$\mathrm{PC}_{\mathbb{R}} \equiv_{\mathrm{W}} \mathrm{PC}^{\Delta}_{[0,1]}$.

Theorem

VCT_2 *is Las Vegas computable with finite mind changes.*

Question

Is VCT_2 *Las Vegas with f.m.m.c. complete (i.e.* $\mathrm{VCT}_2 \equiv_{\mathrm{W}} \mathrm{VCT}_1^{\Delta}$*)?*

We don't know!

Theorem

VCT_2 *is Las Vegas computable with finite mind changes.*

Question

Is VCT_2 *Las Vegas with f.m.m.c. complete (i.e.* $\mathsf{VCT}_2 \equiv_W \mathsf{VCT}_1^{\Delta}$*)?*

We don't know!

Theorem
VCT_2 *is Las Vegas computable with finite mind changes.*

Question
Is VCT_2 *Las Vegas with f.m.m.c. complete (i.e.* $\mathsf{VCT}_2 \equiv_{\mathrm{W}} \mathsf{VCT}_1^{\Delta}$*)?*

We don't know!

Probabilistic Algorithms

What happens when the condition about the failure recognition mechanism does not necessarily apply?

- the successful oracle sets S_p are not necessarily closed
- the successful oracle sets S_p do not need to depend in any uniform way from inputs p.

Probabilistic Algorithms

What happens when the condition about the failure recognition mechanism does not necessarily apply?

- the successful oracle sets S_p are not necessarily closed
- the successful oracle sets S_p do not need to depend in any uniform way from inputs p.

Probabilistic Algorithms

What happens when the condition about the failure recognition mechanism does not necessarily apply?

- the successful oracle sets S_p are not necessarily closed
- the successful oracle sets S_p do not need to depend in any uniform way from inputs p.

Non-uniform computability:

Theorem (Single-valued probabilistic degrees)

Let X be a represented space and let Y be a T_0-space with countable base and standard representation. If a single-valued function $f : X \to Y$ is probabilistic, then it is non uniformly computable, that is f maps computable inputs to computable outputs. Even more $f(x) \leq_r x$ for all $x \in X$.

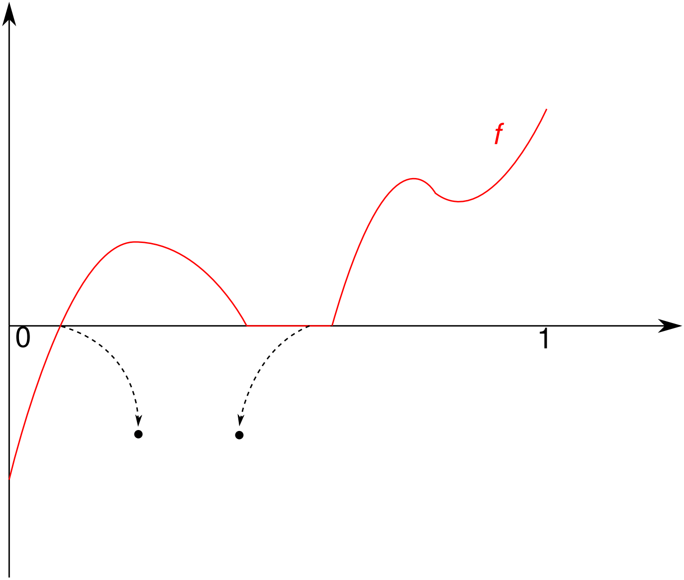
Proposition

f Las Vegas computable $\Longrightarrow$ f probabilistic.

What about the inverse?

Natural probabilistic but not Las Vegas

IVT:



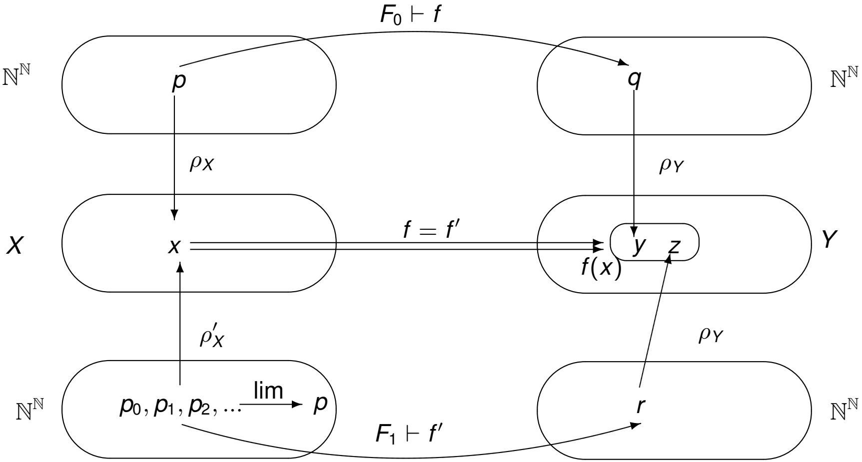


Figure : The derivative jump f' of f.

Theorem

IVT *is probabilistic. In fact* IVT $\leq_{\mathrm{W}}$ WWKL$'$.

Theorem

IVT *is neither Las Vegas computable nor Las Vegas computable with finitely many mind changes.*

Theorem

IVT *is probabilistic. In fact* $\mathrm{IVT} \leq_{\mathrm{W}} \mathrm{WWKL}'$.

Theorem

IVT *is neither Las Vegas computable nor Las Vegas computable with finitely many mind changes.*

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

Conclusions

- We have extended the models of computations to Las Vegas Turing Machines
- We have charachterized Las Vegas computability in terms of Weihrauch reducibility
- We have seen that a version of Vitali Theorem is Las Vegas Computable and another one is Las Vegas Computable with finitely many mind changes
- We have extended the models of computations to probabilistic Turing Machines
- We have seen that (essentially) all single valued probabilistic funtions are non uniformly computable
- We have seen that the Intermediate Value Theorem is probabilistically computable but not Las Vegas computable (even with finitely many mind changes).

V. Brattka, G. G, R. Hölzl: Probabilistic Computability and Choice, *Information and Computation* 242: 249-286 (2015)