Loria - Inria, Nancy (France)
ENS Paris-Saclay (France)

Descriptive complexity on non-Polish spaces

Antonin Callard & Mathieu Hoyrup

Computability in Europe 2020

Loria
Laboratoire lorrain de recherche en informatique et ses applications

Inria

Introduction to represented spaces

Algorithms make sense in the context of represented spaces:

Introduction to represented spaces

Algorithms make sense in the context of represented spaces:

Definition : For X a space, (X, δ) is an (admissibly) represented space if :

$$X$$

Introduction to represented spaces

Algorithms make sense in the context of represented spaces:

Definition : For X a space, (X, δ) is an (admissibly) represented space if :

$$X$$

$$\{0,1\}^{\omega}$$

Introduction to represented spaces

Algorithms make sense in the context of represented spaces:

Definition : For X a space, (X, δ) is an (admissibly) represented space if :

$$\begin{array}{c} X \\ \uparrow \delta \\ \{0,1\}^\omega \end{array}$$

Algorithms make sense in the context of represented spaces:

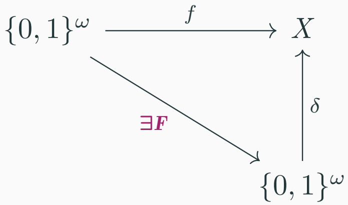
Definition : For X a space, (X, δ) is an (admissibly) represented space if :

$$
\begin{array}{ccc}
\{0,1\}^\omega & \xrightarrow{\quad f \quad} & X \\
 & & \uparrow \delta \\
 & & \{0,1\}^\omega
\end{array}
$$

Introduction to represented spaces

Algorithms make sense in the context of represented spaces:

Definition : For X a space, (X, δ) is an (admissibly) represented space if :



A motivating example

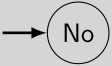
Let $S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$

A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:

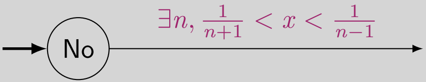


A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:

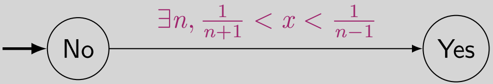


A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:

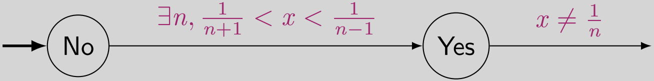


A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:

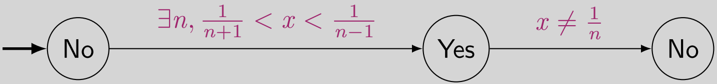


A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:

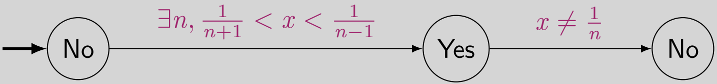


A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:



Topological description:

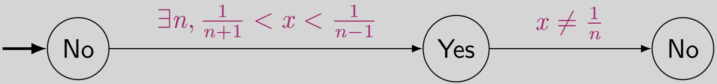
$$S = (0, +\infty)$$

A motivating example

$$\text{Let } S = \left\{ \frac{1}{n} : n \in \mathbb{N} \right\} \subseteq \mathbb{R}$$

Two approaches:

Algorithm:



Topological description:

$$S = (0, +\infty) \setminus \bigcup_{n \in \mathbb{N}} \left(\frac{1}{n+1}, \frac{1}{n} \right)$$

A motivating example

Algorithms and effective topology yield the same results: for any $S \subseteq \mathbb{R}$,

A motivating example

Algorithms and effective topology yield the same results: for any $S \subseteq \mathbb{R}$,

S is decidable with ≤ 2 mind changes (No, Yes, No)

A motivating example

Algorithms and effective topology yield the same results: for any $S \subseteq \mathbb{R}$,

$$S \text{ is decidable with } \leq 2 \text{ mind changes (No, Yes, No)}$$
$$\Longleftrightarrow$$
$$S \text{ is a difference of two effective open sets } (S \in D_2(\mathbb{R}))$$

A motivating example

Algorithms and effective topology yield the same results: for any $S \subseteq \mathbb{R}$,

S is decidable with ≤ 2 mind changes (No, Yes, No)

$\iff$

S is a difference of two effective open sets ($S \in D_2(\mathbb{R})$)

In other words,

$\delta^{-1}(S)$ is a difference of two effective open sets

$\iff$

S is a difference of two effective open sets ($S \in D_2(\mathbb{R})$)

A motivating example

Algorithms and effective topology yield the same results: for any $S \subseteq \mathbb{R}$,

S is decidable with ≤ 2 mind changes (No, Yes, No)

$\iff$

S is a difference of two effective open sets ($S \in D_2(\mathbb{R})$)

In other words,

$\delta^{-1}(S)$ is a difference of two effective open sets

$\iff$

S is a difference of two effective open sets ($S \in D_2(\mathbb{R})$)

Question:

Are those two approaches always equivalent?

Outline

Outline

Formalization of the problem

Defining a notion of complexity

We need to build two notions of complexity for sets $S \subseteq X$:

- One for topological complexity (for $S \subseteq X$).
- One for symbolic complexity (on $\delta^{-1}(S) \subseteq \{0,1\}^\omega$).

Defining a notion of complexity

We need to build two notions of complexity for sets $S \subseteq X$:

- One for topological complexity (for $S \subseteq X$).
- One for symbolic complexity (on $\delta^{-1}(S) \subseteq \{0,1\}^\omega$).

Key idea: semi-decidable problems are similar to (effective) open sets!

Defining a notion of complexity

Defining a notion of complexity

Define algorithms "solving" a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;

Defining a notion of complexity

Define algorithms “solving” a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;
- It outputs a converging sequence of statements $True$ and $False$;

Defining a notion of complexity

Define algorithms "solving" a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;
- It outputs a converging sequence of statements $True$ and $False$;
- The limit is $True$ iff $input \in P$.

Defining a notion of complexity

Define algorithms "solving" a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;
- It outputs a converging sequence of statements $True$ and $False$;
- The limit is $True$ iff $input \in P$.

You can then start building your complexity classes:

- $[\Delta^0_1] \rightarrow$ traditional decision problems.

Defining a notion of complexity

Define algorithms "solving" a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;
- It outputs a converging sequence of statements $True$ and $False$;
- The limit is $True$ iff $input \in P$.

You can then start building your complexity classes:

- $[\Delta_1^0] \rightarrow$ traditional decision problems.
- $[\Sigma_1^0] \rightarrow$ problems P such that:

  There exists an algorithm A of structure **NY**:

  - It first outputs No.
  - It reads its input.
  - It outputs Yes in finite time iff $input \in P$. (otherwise loops)

  (ie. **semi-decidable problems**)

Defining a notion of complexity

Define algorithms "solving" a decision problem P with mind-changes:

- The algorithm runs infinitely and reads a name over time;
- It outputs a converging sequence of statements $True$ and $False$;
- The limit is $True$ iff $input \in P$.

You can then start building your complexity classes:

- $[\Delta^0_1] \rightarrow$ traditional decision problems.
- $[\Sigma^0_1] \rightarrow$ problems P such that:

  There exists an algorithm A of structure **NY**:

  - It first outputs No.
  - It reads its input.
  - It outputs Yes in finite time iff $input \in P$. (otherwise loops)

  (ie. **semi-decidable problems**)
- $[\Pi^0_1]$: problems P verifying $P^c \in [\Sigma^0_1]$, ie. :

  There exists an algorithm A of structure **YN**.

Defining a notion of complexity

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**.

$\Delta^0_1 =$ clopen sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**. $\Delta^0_1 =$ clopen sets.
$[\Sigma^0_1] = [D_1] =$ **NY**. (r.e. problem)

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1] =$ **NY**. (r.e. problem)

$\Delta^0_1 =$ clopen sets.
$\Sigma^0_1 = D_1 =$ open sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1]$ = **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1]$ = **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1]$ = **YN**. (co-r.e. problem)

Δ^0_1 = clopen sets.
$\Sigma^0_1 = D_1$ = open sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1] =$ **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1] =$ **YN**. (co-r.e. problem)

$\Delta^0_1 =$ clopen sets.
$\Sigma^0_1 = D_1 =$ open sets.
$\Pi^0_1 = \check{D}_1 =$ closed sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1]$ = **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1]$ = **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1]$ = **YN**. (co-r.e. problem)
$[D_2]$ = **NYN**.
$[\check{D}_2]$ = **YNY**.

Δ^0_1 = clopen sets.
$\Sigma^0_1 = D_1$ = open sets.
$\Pi^0_1 = \check{D}_1$ = closed sets.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1] =$ **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1] =$ **YN**. (co-r.e. problem)
$[D_2] =$ **NYN**.
$[\check{D}_2] =$ **YNY**.

$\Delta^0_1 =$ clopen sets.
$\Sigma^0_1 = D_1 =$ open sets.
$\Pi^0_1 = \check{D}_1 =$ closed sets.
$D_2 = \{U_1 \setminus U_2\}$.
$\check{D}_2 = \{A : A^{\mathsf{c}} \in D_2\}$.

Defining a notion of complexity

One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

$[\Delta^0_1]$ = **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1]$ = **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1]$ = **YN**. (co-r.e. problem)
$[D_2]$ = **NYN**.
$[\check{D}_2]$ = **YNY**.
$[D_3]$ = **NYNY**.
$[\check{D}_3]$ = **YNYN**.

Δ^0_1 = clopen sets.
$\Sigma^0_1 = D_1$ = open sets.
$\Pi^0_1 = \check{D}_1$ = closed sets.
$D_2 = \{U_1 \setminus U_2\}$.
$\check{D}_2 = \{A : A^{\mathsf{c}} \in D_2\}$.
$D_3 = \{U_1 \setminus (U_2 \setminus U_3)\}$.
$\check{D}_3 = \{A : A^{\mathsf{c}} \in D_2\}$.

Defining a notion of complexity

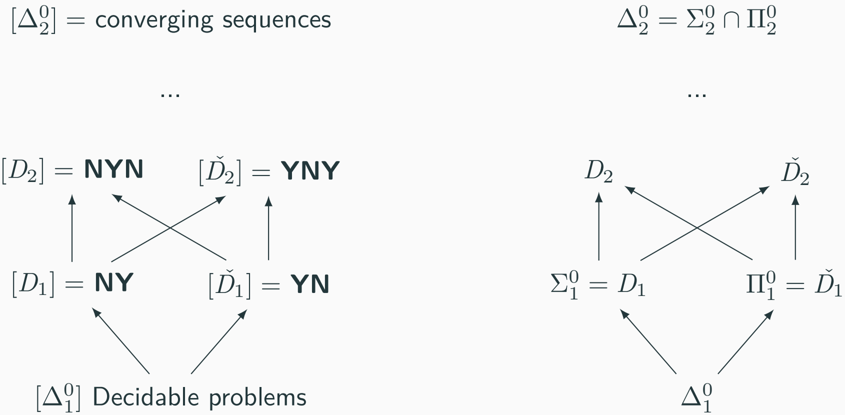
One defines two hierarchies with high degree of symmetry:

- Bound on the number of mind-changes.
- Number of differences of open sets.

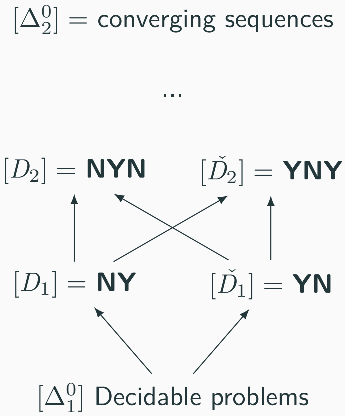
$[\Delta^0_1] =$ **decidable**, ie. **Y** or **N**.
$[\Sigma^0_1] = [D_1] =$ **NY**. (r.e. problem)
$[\Pi^0_1] = [\check{D}_1] =$ **YN**. (co-r.e. problem)
$[D_2] =$ **NYN**.
$[\check{D}_2] =$ **YNY**.
$[D_3] =$ **NYNY**.
$[\check{D}_3] =$ **YNYN**.
etc...
$[\Delta^0_2] =$ **converging sequences**.

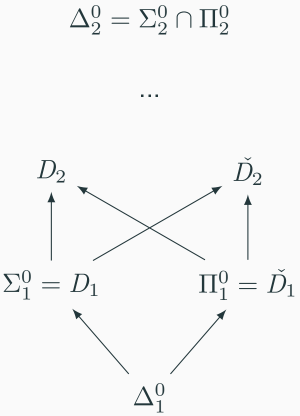
$\Delta^0_1 =$ clopen sets.
$\Sigma^0_1 = D_1 =$ open sets.
$\Pi^0_1 = \check{D}_1 =$ closed sets.
$D_2 = \{U_1 \setminus U_2\}$.
$\check{D}_2 = \{A : A^c \in D_2\}$.
$D_3 = \{U_1 \setminus (U_2 \setminus U_3)\}$.
$\check{D}_3 = \{A : A^c \in D_2\}$.
etc...
$\Delta^0_2 = \Sigma^0_2 \cap \Pi^0_2$,
where $\Sigma^0_2 = \{\bigcup_{n \in \mathbb{N}} U_n \setminus U'_n\}$.

Reformulation of the problem



Reformulation of the problem





For $\Gamma \in \{D_\eta : \eta < \omega_1\}$, when does one have $\Gamma = [\Gamma]$?

Countably-based spaces

Countably-based spaces: matching of the two hierarchies

cb_0 spaces: countably-based (a countable number of open sets describes the topology) + T_0 axiom

Countably-based spaces: matching of the two hierarchies

cb_0 spaces: countably-based (a countable number of open sets describes the topology) + T_0 axiom

Theorem 2.1: M. De Brecht, 2012

For (X, δ) a countably-based represented space,

$$S \in \Gamma \iff S \in [\Gamma]$$

Countably-based spaces: matching of the two hierarchies

cb_0 spaces: countably-based (a countable number of open sets describes the topology) $+$ T_0 axiom

[New] - Theorem 2.2: Hierarchies coincide in an effective way

For (X, δ) a countably-based represented space,

$$S \in \Gamma \iff S \in [\Gamma]$$

in an effective and uniform way.

Countably-based: about hardness

Countably-based: about hardness

In (traditional) complexity theory, one defines the notion of "hardness" to prove that a problem is "harder than" another one.

(Roughly) For A and B problems, $A \leq B$ if there exists $f : A \mapsto B$.

Countably-based: about hardness

In (traditional) complexity theory, one defines the notion of "hardness" to prove that a problem is "harder than" another one.

(Roughly) For A and B problems, $A \leq B$ if there exists $f : A \mapsto B$.

We proceed in the same way here, and define:

For $A \subseteq X$ and $B \subseteq Y$, $A \leq_W$ B if there exists $f : X \mapsto Y$ such that $f^{-1}(B) = A$.

Countably-based: about hardness

In (traditional) complexity theory, one defines the notion of "hardness" to prove that a problem is "harder than" another one.

(Roughly) For A and B problems, $A \leq B$ if there exists $f : A \mapsto B$.

We proceed in the same way here, and define:

For $A \subseteq X$ and $B \subseteq Y$, $A \leq_W$ B if there exists $f : X \mapsto Y$ such that $f^{-1}(B) = A$.
For a set S, S is Γ-hard if for any $A \in \Gamma(\{0,1\}^\omega)$, $A \leq_W S$.

Countably-based: about hardness

In (traditional) complexity theory, one defines the notion of "hardness" to prove that a problem is "harder than" another one.

(Roughly) For A and B problems, $A \leq B$ if there exists $f : A \mapsto B$.

We proceed in the same way here, and define:

For $A \subseteq X$ and $B \subseteq Y$, $A \leq_W$ B if there exists $f : X \mapsto Y$ such that $f^{-1}(B) = A$.
For a set S, S is Γ-hard if for any $A \in \Gamma(\{0,1\}^\omega)$, $A \leq_W S$.

In cb_0 spaces, if $\check{\Gamma} = \{P : P^c \in \Gamma\}$,then:

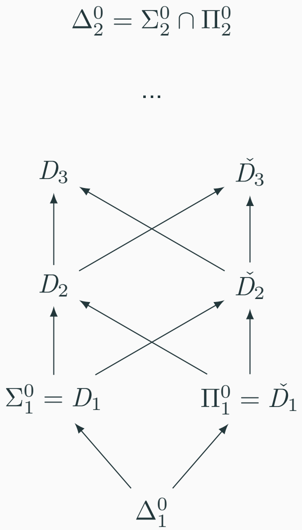
[New] - Theorem 2.7: Hardness-criterion

$$S \notin D_\eta \iff S \text{ is } \check{D}_\eta\text{-hard.}$$

Countably-based: about hardness

Theorem 2.8: Hardness-criterion

$$S \notin D_\eta \iff S \text{ is } \check{D}_\eta\text{-hard.}$$



Non countably-based spaces

Real polynomials

Real polynomials

Consider the space of real polynomials $\mathbb{R}[X]$ equipped with the CoPolish topology:

Real polynomials

Consider the space of real polynomials $\mathbb{R}[X]$ equipped with the CoPolish topology:

$$\mathcal{O}(\mathbb{R}[X]) = \{O \subseteq \mathbb{R}[X] : \forall n, O \text{ is open in } \mathbb{R}_n[X]\}$$

Real polynomials

Consider the space of real polynomials $\mathbb{R}[X]$ equipped with the CoPolish topology:

$$\mathcal{O}(\mathbb{R}[X]) = \{O \subseteq \mathbb{R}[X] : \forall n, O \text{ is open in } \mathbb{R}_n[X]\}$$

Represent a polynomial $P \in \mathbb{R}[X]$ by :

- Some $n \geq \deg(P)$.

Real polynomials

Consider the space of real polynomials $\mathbb{R}[X]$ equipped with the CoPolish topology:

$$\mathcal{O}(\mathbb{R}[X]) = \{O \subseteq \mathbb{R}[X] : \forall n, O \text{ is open in } \mathbb{R}_n[X]\}$$

Represent a polynomial $P \in \mathbb{R}[X]$ by :

- Some $n \geq \deg(P)$.
- The coefficients $p_0, ..., p_n$ such that $P = p_0 + p_1 X + ... + p_n X^n$.

Real polynomials

Consider the space of real polynomials $\mathbb{R}[X]$ equipped with the CoPolish topology:

$$\mathcal{O}(\mathbb{R}[X]) = \{O \subseteq \mathbb{R}[X] : \forall n, O \text{ is open in } \mathbb{R}_n[X]\}$$

Represent a polynomial $P \in \mathbb{R}[X]$ by :

- Some $n \geq \deg(P)$.
- The coefficients $p_0, ..., p_n$ such that $P = p_0 + p_1 X + ... + p_n X^n$.

It is an admissible representation of $\mathbb{R}[X]$.

Complexity of S_1

Property 3.1: Complexity of S_1 (Part 1)

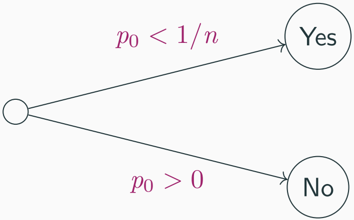
$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \in [D_2]$$

Complexity of S_1

Property 3.1: Complexity of S_1 (Part 1)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \in [D_2]$$

Proof :

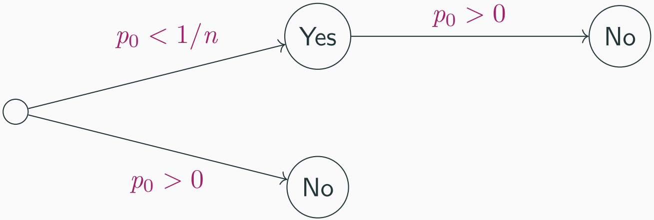


Complexity of S_1

Property 3.1: Complexity of S_1 (Part 1)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \in [D_2]$$

Proof :

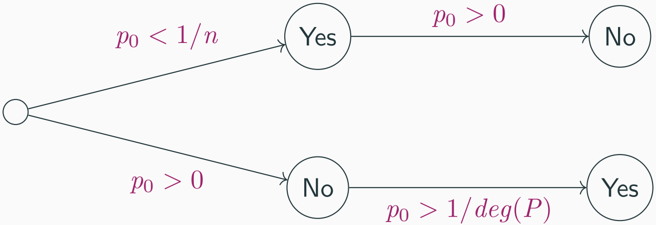


Complexity of S_1

Property 3.1: Complexity of S_1 (Part 1)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \in [D_2]$$

Proof :



Property 3.2: Complexity of S_1 (Part 2)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \notin D_2$$

(ie. S_1 is not a difference of two open sets)

Property 3.2: Complexity of S_1 (Part 2)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \notin D_2$$

(ie. S_1 is not a difference of two open sets)

Proof:

$$\frac{1}{n} + \frac{X^{n+1}}{p} \quad (\in S_1)$$

Complexity of S_1

Property 3.2: Complexity of S_1 (Part 2)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \notin D_2$$

(ie. S_1 is not a difference of two open sets)

Proof:

$$\frac{1}{n} + \frac{X^{n+1}}{p} \ (\in S_1) \xrightarrow[p \mapsto +\infty]{} \frac{1}{n} \ (\notin S_1)$$

Property 3.2: Complexity of S_1 (Part 2)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \notin D_2$$

(ie. S_1 is not a difference of two open sets)

Proof:

$$\frac{1}{n} + \frac{X^{n+1}}{p} \ (\in S_1) \ \xrightarrow[p \mapsto +\infty]{} \ \frac{1}{n} \ (\notin S_1) \ \xrightarrow[n \mapsto +\infty]{} \ 0 \ (\in S_1)$$

Complexity of S_1

Property 3.2: Complexity of S_1 (Part 2)

$$S_1 = \left\{ P \in \mathbb{R}[X] : p_0 = 0 \text{ or } p_0 > \frac{1}{\deg(P)} \right\} \notin D_2$$

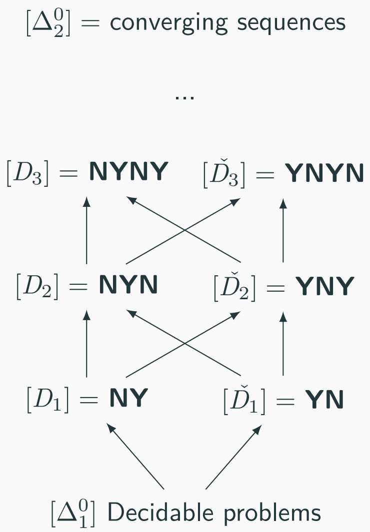
(ie. S_1 is not a difference of two open sets)

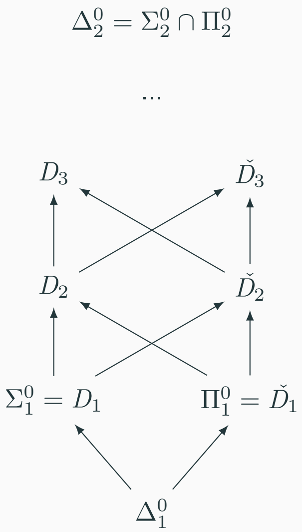
Proof:

$$\frac{1}{n} + \frac{X^{n+1}}{p} \ (\in S_1) \xrightarrow[p \mapsto +\infty]{} \frac{1}{n} \ (\notin S_1) \xrightarrow[n \mapsto +\infty]{} 0 \ (\in S_1)$$

Conclusion : $S_1 \in [D_2]$ but $S_1 \notin D_2$

Complexity of S_1





To go even further... Introduce S_2.

$$S_2 = \left\{ \frac{1}{k_1} + \frac{X^{k_1}}{k_2} + ... + \frac{X^{k_n}}{k_{n+1}} : \exists n, k_1 < ... < k_{n+1}, k_n \text{ even} \right\}$$

[New] - Theorem 3.3: Complexity of S_2

To go even further... Introduce S_2.

$$S_2 = \left\{ \frac{1}{k_1} + \frac{X^{k_1}}{k_2} + \ldots + \frac{X^{k_n}}{k_{n+1}} : \exists n, k_1 < \ldots < k_{n+1}, k_n \text{ even} \right\}$$

[New] - Theorem 3.3: Complexity of S_2

1. $S_2 \in [D_2]$.

To go even further... Introduce S_2.

$$S_2 = \left\{ \frac{1}{k_1} + \frac{X^{k_1}}{k_2} + \dots + \frac{X^{k_n}}{k_{n+1}} : \exists n, k_1 < \dots < k_{n+1}, k_n \text{ even} \right\}$$

[New] - Theorem 3.3: Complexity of S_2

1. $S_2 \in [D_2]$.
2. But $\forall \eta < \omega_1, S_2 \notin D_\eta$.

To go even further... Introduce S_2.

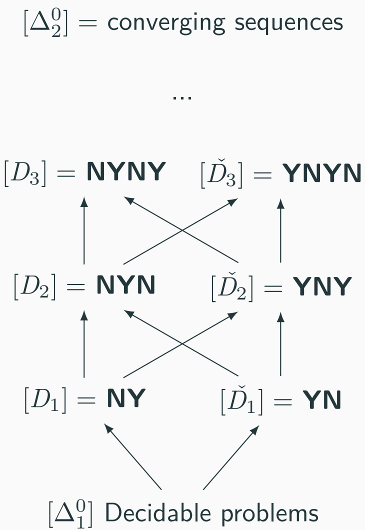
$$S_2 = \left\{ \frac{1}{k_1} + \frac{X^{k_1}}{k_2} + \ldots + \frac{X^{k_n}}{k_{n+1}} : \exists n, k_1 < \ldots < k_{n+1}, k_n \text{ even} \right\}$$

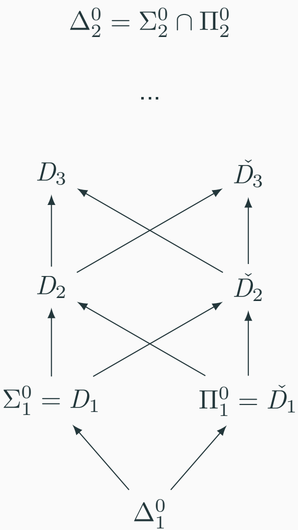
[New] - Theorem 3.3: Complexity of S_2

1. $S_2 \in [D_2]$.
2. But $\forall \eta < \omega_1, S_2 \notin D_\eta$.

Conclusion : $S_2 \in [D_2]$ but $S_2 \notin \bigcup_{\eta < \omega_1} D_\eta$

Complexity of S_2





What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

continuity sequential continuity

What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

continuity	sequential continuity
compactness	sequential compactness

What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

continuity	sequential continuity
compactness	sequential compactness
closure	sequential closure

What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

continuity	sequential continuity
compactness	sequential compactness
closure	sequential closure

Why do we think that?

What happened?

Why is this happening?

Guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

continuity	sequential continuity
compactness	sequential compactness
closure	sequential closure

Why do we think that?

[New] - Theorem 3.4: $[\Gamma]$ depends on sequentiality

For a coPolish space X, $1.$ and $2.$ are equivalent.

1. Closure and sequential closure coincide on X.
2. For every $n < \omega$, $A \in [D_n]$ iff $A \in D_n$.

Consequences - About hardness

Recall that the definition of Γ-hardness involves $\{0,1\}^\omega$.

Consequences - About hardness

Recall that the definition of Γ-hardness involves $\{0,1\}^\omega$.

The notion of **hardness captures the symbolic complexity**,
and *not* the topological one!

Consequences - About hardness

Recall that the definition of Γ-hardness involves $\{0,1\}^\omega$.

The notion of **hardness captures the symbolic complexity**,
and *not* the topological one!

For example:

[New] Theorem 3.5: Countably-based spaces

$$A \notin D_\eta \iff A \text{ is } \check{D}_\eta\text{-hard.}$$

Consequences - About hardness

Recall that the definition of Γ-hardness involves $\{0,1\}^\omega$.

The notion of **hardness captures the symbolic complexity**,
and *not* the topological one!

For example:

[New] Theorem 3.6: co-Polish spaces

$$A \notin [D_\eta] \iff A \text{ is } \check{D}_\eta\text{-hard.}$$

Consequences - About hardness

Recall that the definition of Γ-hardness involves $\{0,1\}^\omega$.

The notion of **hardness captures the symbolic complexity**,
and *not* the topological one!

For example:

[New] Theorem 3.6: co-Polish spaces

$$A \notin [D_\eta] \iff A \text{ is } \check{D}_\eta\text{-hard.}$$

DST works better with symbolic complexity (and represented spaces)
rather than with topology.

Conclusion

Conclusion

Let us recap:

- In countably-based spaces: (computable) equivalence of complexities Γ and $[\Gamma]$.
- In $\mathbb{R}[X]$: the equivalence is broken

Conclusion

Let us recap:

- In countably-based spaces: (computable) equivalence of complexities Γ and $[\Gamma]$.
- In $\mathbb{R}[X]$: the equivalence is broken (and symbolic complexity is better!)

Conclusion

Let us recap:

- In countably-based spaces: (computable) equivalence of complexities Γ and $[\Gamma]$.
- In $\mathbb{R}[X]$: the equivalence is broken (and symbolic complexity is better!)

Open questions: Why is the equivalence broken? On which spaces does it happen, at which complexity levels, and why?

Conclusion

Let us recap:

- In countably-based spaces: (computable) equivalence of complexities Γ and $[\Gamma]$.
- In $\mathbb{R}[X]$: the equivalence is broken (and symbolic complexity is better!)

Open questions: Why is the equivalence broken? On which spaces does it happen, at which complexity levels, and why?

Our guess: The difference is related to the mismatch between **topological** and **sequential** aspects.

Questions?