Hierarchies of ω-languages

Jérémie Cabessa

University Joseph Fourier – Grenoble 1

13 February 2009

Outline

Outline

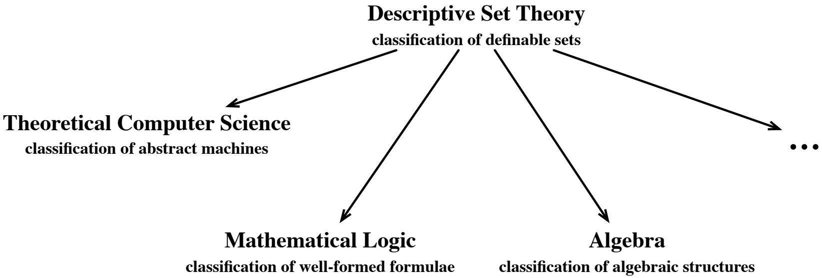
Outline

Outline

Outline

Outline

Descriptive Set Theory provides a general framework for classification problems.



ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An *ω-language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An *ω-language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An *ω-language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An *ω-language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An ω-*language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An ω-*language* is a set of infinite words.

ω-languages

- A *language* is a set of words.
- An ω-language is a set of infinite words.

Definition

- An *alphabet* is simply a set whose elements are called *letters*.
- A *word* is a sequence of letters of given alphabet. An *infinite word* is an infinite sequence of letters.
- A *language* is a set of words.
- An ω-*language* is a set of infinite words.

Example

- We will usually consider the alphabet $A = \{0, 1\}$.
- Words on A: $w_0 = 01101$, $w_1 = 11101$, $\ldots$
- Infinite words on A: $\alpha = 0000\cdots$, $\beta = 1001010\cdots$, $\ldots$
- Language on A: $L = \{00, 011, 10, 1101\}$.
- ω-Language on A: $L = \{0000\cdots, 100\cdots, 1100\cdots\}$.

Example

- We will usually consider the alphabet $A = \{0, 1\}$.
- Words on A: $w_0 = 01101$, $w_1 = 11101$, ...
- Infinite words on A: $\alpha = 0000\cdots$, $\beta = 1001010\cdots$, ...
- Language on A: $L = \{00, 011, 10, 1101\}$.
- ω-Language on A: $L = \{0000\cdots, 100\cdots, 1100\cdots\}$.

Example

- We will usually consider the alphabet $A = \{0, 1\}$.
- Words on A: $w_0 = 01101$, $w_1 = 11101$, ...
- Infinite words on A: $\alpha = 0000\cdots$, $\beta = 1001010\cdots$, ...
- Language on A: $L = \{00, 011, 10, 1101\}$.
- ω-Language on A: $L = \{0000\cdots, 100\cdots, 1100\cdots\}$.

Example

- We will usually consider the alphabet $A = \{0, 1\}$.
- Words on A: $w_0 = 01101$, $w_1 = 11101$, ...
- Infinite words on A: $\alpha = 0000\cdots$, $\beta = 1001010\cdots$, ...
- Language on A: $L = \{00, 011, 10, 1101\}$.
- ω-Language on A: $L = \{0000\cdots, 100\cdots, 1100\cdots\}$.

Example

- We will usually consider the alphabet $A = \{0, 1\}$.
- Words on A: $w_0 = 01101$, $w_1 = 11101$, ...
- Infinite words on A: $\alpha = 0000\cdots$, $\beta = 1001010\cdots$, ...
- Language on A: $L = \{00, 011, 10, 1101\}$.
- ω-Language on A: $L = \{0000\cdots, 100\cdots, 1100\cdots\}$.

An abstract machine $\mathcal{M}$ can be identified with the languages $L(\mathcal{A})$ that it recognizes.

Example

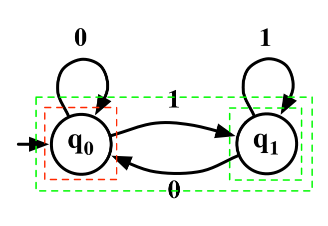
The following Muller automaton $\mathcal{A}$ can be identified with the ω-language $L(\mathcal{A})$ of infinite words with infinitely many 1's.

Hence abstract machines can be considered as specific ω-languages!

An abstract machine $\mathcal{M}$ can be identified with the languages $L(\mathcal{A})$ that it recognizes.

Example

The following Muller automaton $\mathcal{A}$ can be identified with the ω-language $L(\mathcal{A})$ of infinite words with infinitely many 1's.

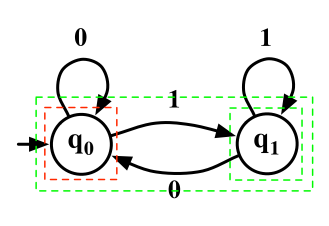


Hence abstract machines can be considered as specific ω-languages!

An abstract machine $\mathcal{M}$ can be identified with the languages $L(\mathcal{A})$ that it recognizes.

Example

The following Muller automaton $\mathcal{A}$ can be identified with the ω-language $L(\mathcal{A})$ of infinite words with infinitely many 1's.



Hence abstract machines can be considered as specific ω-languages!

A decision problem P can be identified the languages L_P of all its solutions.

Example

For instance, consider the following decision problem and language:

- P: Given a finite graph G, is G acyclic or not?
- L_P: the set of all acyclic finite graphs (language of solutions).

Then the question "is G is acyclic?" **is equivalent** to the question "does $G \in L_P$?".

Hence decision problems can be considered as specific languages!

A decision problem P can be identified the languages L_P of all its solutions.

Example

For instance, consider the following decision problem and language:

- P: Given a finite graph G, is G acyclic or not?
- L_P: the set of all acyclic finite graphs (language of solutions).

Then the question “is G is acyclic?” **is equivalent** to the question “does $G \in L_P$?”.

Hence decision problems can be considered as specific languages!

A decision problem P can be identified the languages L_P of all its solutions.

Example

For instance, consider the following decision problem and language:

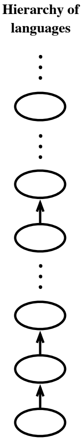
- P: Given a finite graph G, is G acyclic or not?
- L_P: the set of all acyclic finite graphs (language of solutions).

Then the question "is G is acyclic?" **is equivalent** to the question "does $G \in L_P$?".

Hence decision problems can be considered as specific languages!

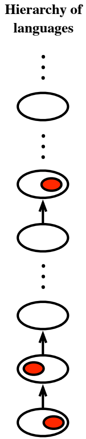
Therefore a classification of languages induces in particular:

- A classification of abstract machines;
- A classification of decision problems;



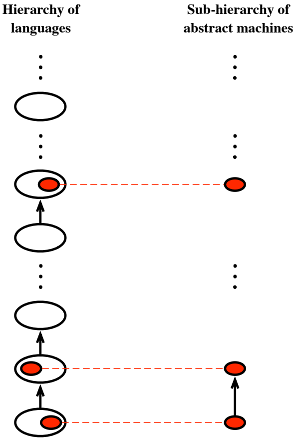
Therefore a classification of languages induces in particular:

- A classification of abstract machines;
- A classification of decision problems;



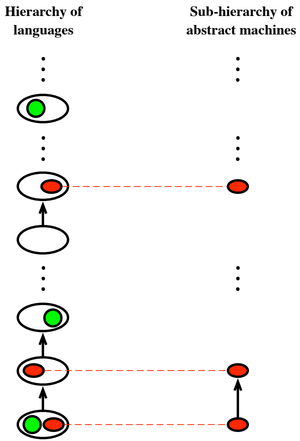
Therefore a classification of languages induces in particular:

- A classification of abstract machines;
- A classification of decision problems;



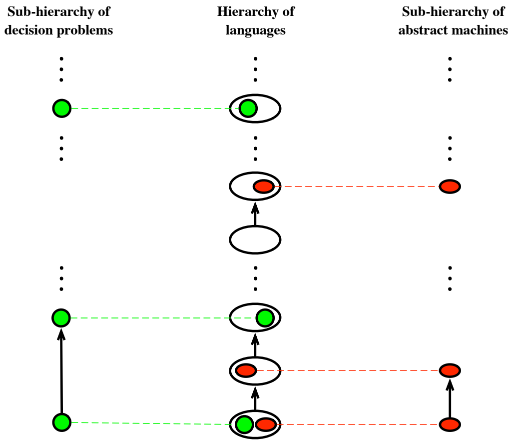
Therefore a classification of languages induces in particular:

- A classification of abstract machines;
- A classification of decision problems;



Therefore a classification of languages induces in particular:

- A classification of abstract machines;
- A classification of decision problems;



In fact, a given classification of languages induces a classification of “anything that can be coded by a language”!

Therefore descriptive set theory provides a general approach of classification problems.

The Cantor space

The *Cantor space* is the space of all infinite words of bits.

Definition

Consider the alphabet $A = \{0, 1\}$. The *Cantor space* $\mathcal{C}$ is the set of all infinite words on A.

$$
\begin{aligned}
\mathcal{C} = \{ & 000000\cdots, \\
& 100000\cdots, \\
& 110000\cdots, \\
& 110000\cdots, \\
& \vdots \\
& \ldots \}
\end{aligned}
$$

The Cantor space

The *Cantor space* is the space of all infinite words of bits.

Definition

Consider the alphabet $A = \{0, 1\}$. The *Cantor space* $\mathcal{C}$ is the set of all infinite words on A.

$$\begin{aligned}\mathcal{C} = \{ & 000000\cdots, \\ & 100000\cdots, \\ & 110000\cdots, \\ & 110000\cdots, \\ & \vdots \\ & \ldots\}\end{aligned}$$

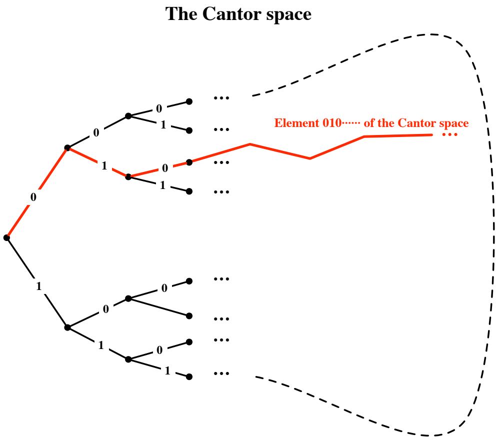
Lemma

The Cantor space $\mathcal{C}$ is uncountable. More precisely, $card(\mathcal{C}) = card(\mathbb{R})$.

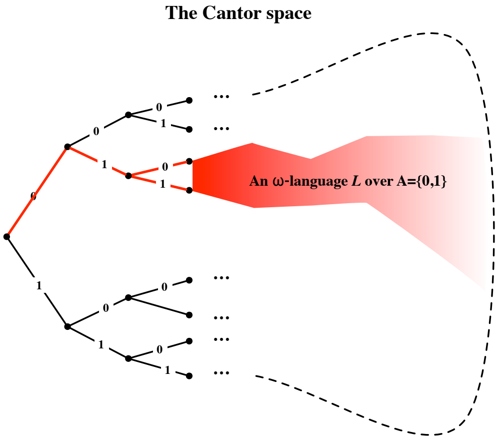
Proof.

A diagonal argument. $\square$

Elements of the Cantor space $\mathcal{C}$ are infinite words of bits.
Elements of the Cantor space $\mathcal{C}$ “are” the infinite branches of the infinite binary tree.

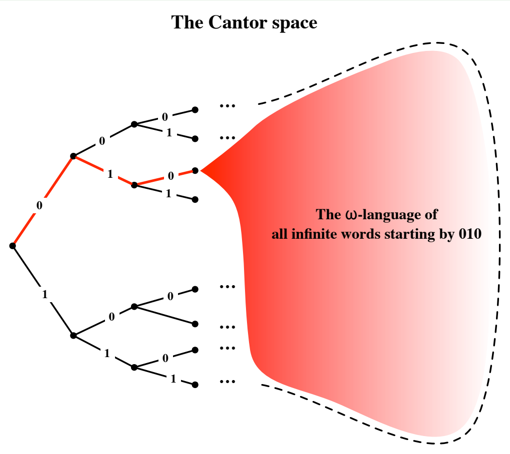


Subsets of the Cantor space are the ω-languages over $A = \{0, 1\}$.
Subsets of the Cantor space $\mathcal{C}$ "are" set of infinite branches of the infinite binary tree.



Example

The ω-languages of all infinite words starting by 010.



Summary:

- Elements of $\mathcal{C}$ are infinite words of bits.
- Subsets of $\mathcal{C}$ are ω-languages over bits.

Summary:

- Elements of $\mathcal{C}$ are infinite words of bits.
- Subsets of $\mathcal{C}$ are ω-languages over bits.

Summary:

- Elements of $\mathcal{C}$ are infinite words of bits.
- Subsets of $\mathcal{C}$ are ω-languages over bits.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

- The *basic open sets* are the prefix ω-languages i.e.

  B is a *basic open set*
  iff
  B contains all infinite words
  that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

- The *basic open sets* are the prefix ω-languages i.e.

  B is a *basic open set*
  iff
  B contains all infinite words
  that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

- The *basic open sets* are the prefix ω-languages i.e.

B is a *basic open set*
iff
B contains all infinite words
that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

- The *basic open sets* are the prefix ω-languages i.e.

  B is a *basic open set*
  iff
  B contains all infinite words
  that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

- The *basic open sets* are the prefix ω-languages i.e.

  B is a *basic open set*
  iff
  B contains all infinite words
  that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

The Borel hierarchy

The *Borel hierarchy* is the most common classification of subsets of a given topological space. In our case, we first equip the Cantor space $\mathcal{C}$ with the following topology:

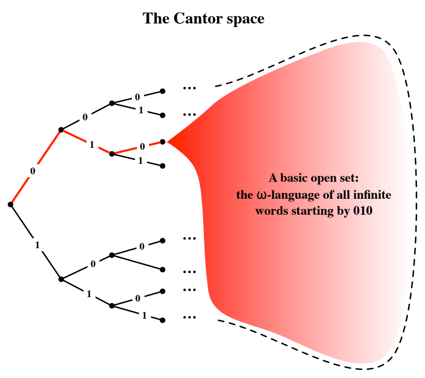
- The *basic open sets* are the prefix ω-languages i.e.

  B is a *basic open set*
  iff
  B contains all infinite words
  that begin with a given prefix p.

- The *open sets* are the arbitrary unions of basic open sets.
- The *closed sets* are the complements of open sets.

Example

A basic open set: the ω-language consisting of all infinite words beginning with 010.



Definition

The class of *Borel sets* of $\mathcal{C}$ is the σ-algebra generated by the open sets, i.e. *Borel subsets* of $\mathcal{C}$ are the ω-languages that can be obtained from open sets by the operations of countable union and complementation.

Definition

The class of *Borel sets* of $\mathcal{C}$ is the σ-algebra generated by the open sets, i.e. *Borel subsets* of $\mathcal{C}$ are the ω-languages that can be obtained from open sets by the operations of countable union and complementation.

The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\boldsymbol{\Sigma}^0_1 = \{A \subseteq \mathcal{C} : A \text{ is open}\}$
- $\boldsymbol{\Sigma}^0_{n+1} = \{\bigcup_{n\in\mathbb{N}} A_n : A_n \in \boldsymbol{\Sigma}^0_n\}$
- $\boldsymbol{\Pi}^0_n = \{A : A^{\complement} \in \boldsymbol{\Sigma}^0_n\}$
- $\boldsymbol{\Delta}^0_n = \{A : A \in \boldsymbol{\Sigma}^0_n \cap \boldsymbol{\Pi}^0_n\}$

The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\mathbf{\Sigma}_1^0 = \{A \subseteq \mathcal{C} : A \text{ is open}\}$
- $\mathbf{\Sigma}_{n+1}^0 = \{\bigcup_{n\in\mathbb{N}} A_n : A_n \in \mathbf{\Sigma}_n^0\}$
- $\mathbf{\Pi}_n^0 = \{A : A^\complement \in \mathbf{\Sigma}_n^0\}$
- $\mathbf{\Delta}_n^0 = \{A : A \in \mathbf{\Sigma}_n^0 \cap \mathbf{\Pi}_n^0\}$

The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\mathbf{\Sigma}_1^0 = \{A \subseteq \mathcal{C} : A$ is open$\}$
- $\mathbf{\Sigma}_{n+1}^0 = \{\bigcup_{n \in \mathbb{N}} A_n : A_n \in \mathbf{\Sigma}_n^0\}$
- $\mathbf{\Pi}_n^0 = \{A : A^\complement \in \mathbf{\Sigma}_n^0\}$
- $\mathbf{\Delta}_n^0 = \{A : A \in \mathbf{\Sigma}_n^0 \cap \mathbf{\Pi}_n^0\}$

The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\mathbf{\Sigma}^0_1 = \{A \subseteq \mathcal{C} : A \text{ is open}\}$
- $\mathbf{\Sigma}^0_{n+1} = \{\bigcup_{n \in \mathbb{N}} A_n : A_n \in \mathbf{\Sigma}^0_n\}$
- $\mathbf{\Pi}^0_n = \{A : A^\complement \in \mathbf{\Sigma}^0_n\}$
- $\mathbf{\Delta}^0_n = \{A : A \in \mathbf{\Sigma}^0_n \cap \mathbf{\Pi}^0_n\}$

The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\boldsymbol{\Sigma}^0_1 = \{A \subseteq \mathcal{C} : A \text{ is open}\}$
- $\boldsymbol{\Sigma}^0_{n+1} = \{\bigcup_{n\in\mathbb{N}} A_n : A_n \in \boldsymbol{\Sigma}^0_n\}$
- $\boldsymbol{\Pi}^0_n = \{A : A^{\complement} \in \boldsymbol{\Sigma}^0_n\}$
- $\boldsymbol{\Delta}^0_n = \{A : A \in \boldsymbol{\Sigma}^0_n \cap \boldsymbol{\Pi}^0_n\}$

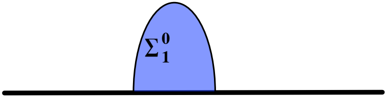
The *Borel hierarchy* classifies Borel sets according to how many times the operations of countable unions and complementation appear in their definitions.

Definition

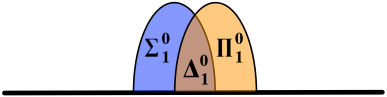
The *Borel finite levels* of the *Borel hierarchy* are defined by induction on n as follows:

- $\mathbf{\Sigma}^0_1 = \{A \subseteq \mathcal{C} : A \text{ is open}\}$
- $\mathbf{\Sigma}^0_{n+1} = \{\bigcup_{n \in \mathbb{N}} A_n : A_n \in \mathbf{\Sigma}^0_n\}$
- $\mathbf{\Pi}^0_n = \{A : A^\complement \in \mathbf{\Sigma}^0_n\}$
- $\mathbf{\Delta}^0_n = \{A : A \in \mathbf{\Sigma}^0_n \cap \mathbf{\Pi}^0_n\}$

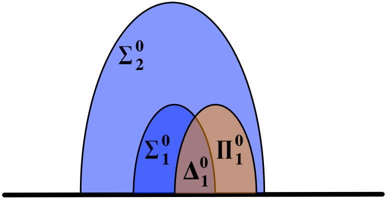
The Borel Hierarchy



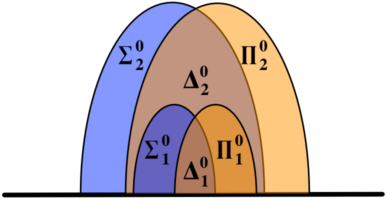
The Borel Hierarchy



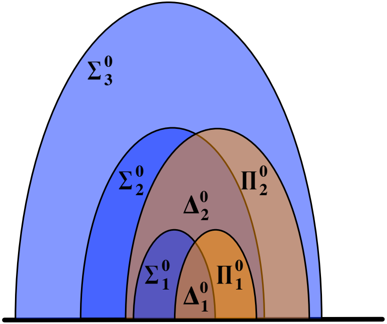
The Borel Hierarchy



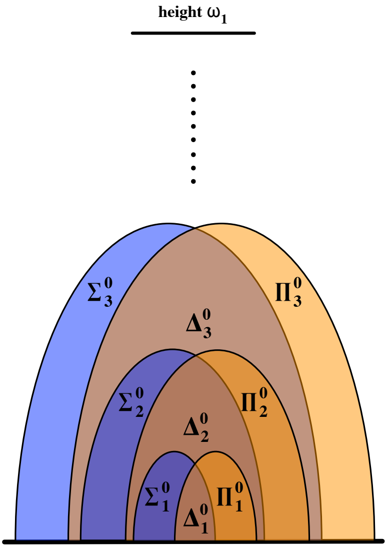
The Borel Hierarchy



The Borel Hierarchy



The Borel Hierarchy



The Borel Hierarchy

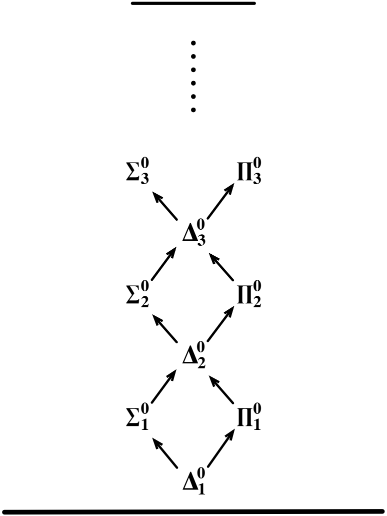
height ω_1

$$\vdots$$

$$
\begin{tikzcd}[column sep=small, row sep=small]
\boldsymbol{\Sigma}^0_3 & & \boldsymbol{\Pi}^0_3 \\
& \boldsymbol{\Delta}^0_3 \arrow[ul, phantom, "\subsetneq" sloped] \arrow[ur, phantom, "\subsetneq" sloped] & \\
\boldsymbol{\Sigma}^0_2 \arrow[ur, phantom, "\subsetneq" sloped] & & \boldsymbol{\Pi}^0_2 \arrow[ul, phantom, "\subsetneq" sloped] \\
& \boldsymbol{\Delta}^0_2 \arrow[ul, phantom, "\subsetneq" sloped] \arrow[ur, phantom, "\subsetneq" sloped] & \\
\boldsymbol{\Sigma}^0_1 \arrow[ur, phantom, "\subsetneq" sloped] & & \boldsymbol{\Pi}^0_1 \arrow[ul, phantom, "\subsetneq" sloped] \\
& \boldsymbol{\Delta}^0_1 \arrow[ul, phantom, "\subsetneq" sloped] \arrow[ur, phantom, "\subsetneq" sloped] &
\end{tikzcd}
$$

The Borel Hierarchy

height ω_1



The Wadge hierarchy

The Wadge hierarchy a drastic refinement of the Borel hierarchy.

Definition

Let A and B subsets of the Cantor space $\mathcal{C}$ (i.e. two ω-languages). The *continuous reduction* $\leq_W$ is defined by:

$$\begin{array}{lll} A \leq_W B & \text{iff} & \text{there exists } f \text{ continuous s.t. } A = f^{-1}(B) \\ & \text{iff} & \text{there exists } f \text{ continuous s.t. } x \in A \Leftrightarrow f(x) \in B \end{array}$$

Then as usual

$$\begin{array}{lll} A <_W B & \text{iff} & A \leq_W B \text{ and } B \not\leq_W A \\ A \equiv_W B & \text{iff} & A \leq_W B \text{ and } B \leq_W A \end{array}$$

The Wadge hierarchy

The Wadge hierarchy a drastic refinement of the Borel hierarchy.

Definition

Let A and B subsets of the Cantor space $\mathcal{C}$ (i.e. two ω-languages). The *continuous reduction* $\leq_W$ is defined by:

$$A \leq_W B \quad \text{iff} \quad \text{there exists } f \text{ continuous s.t. } A = f^{-1}(B)$$
$$\text{iff} \quad \text{there exists } f \text{ continuous s.t. } x \in A \Leftrightarrow f(x) \in B$$

Then as usual

$$A <_W B \quad \text{iff} \quad A \leq_W B \text{ and } B \not\leq_W A$$
$$A \equiv_W B \quad \text{iff} \quad A \leq_W B \text{ and } B \leq_W A$$

The Wadge hierarchy

The Wadge hierarchy a drastic refinement of the Borel hierarchy.

Definition

Let A and B subsets of the Cantor space $\mathcal{C}$ (i.e. two ω-languages). The *continuous reduction* $\leq_W$ is defined by:

$$\begin{array}{lll} A \leq_W B & \text{iff} & \text{there exists } f \text{ continuous s.t. } A = f^{-1}(B) \\ & \text{iff} & \text{there exists } f \text{ continuous s.t. } x \in A \Leftrightarrow f(x) \in B \end{array}$$

Then as usual

$$\begin{array}{lll} A <_W B & \text{iff} & A \leq_W B \text{ and } B \not\leq_W A \\ A \equiv_W B & \text{iff} & A \leq_W B \text{ and } B \leq_W A \end{array}$$

The Wadge hierarchy

The Wadge hierarchy a drastic refinement of the Borel hierarchy.

Definition

Let A and B subsets of the Cantor space $\mathcal{C}$ (i.e. two ω-languages). The *continuous reduction* $\leq_W$ is defined by:

$$\begin{array}{lll} A \leq_W B & \text{iff} & \text{there exists } f \text{ continuous s.t. } A = f^{-1}(B) \\ & \text{iff} & \text{there exists } f \text{ continuous s.t. } x \in A \Leftrightarrow f(x) \in B \end{array}$$

Then as usual

$$\begin{array}{lll} A <_W B & \text{iff} & A \leq_W B \text{ and } B \not\leq_W A \\ A \equiv_W B & \text{iff} & A \leq_W B \text{ and } B \leq_W A \end{array}$$

$A \leq_W B$ means intuitively than A is "less complicated" than B. Indeed we have ...

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \text{ for some } f \text{ continuous}$$

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is "less complicated" than B. Indeed we have ...

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \text{ for some } f \text{ continuous}$$

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is "less complicated" than B. Indeed we have . . .

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \ \text{ for some } f \text{ continuous}$$

Assume that for any element x, you know whether $x \in B$ or not

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is “less complicated” than B. Indeed we have …

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \text{ for some } f \text{ continuous}$$

Does a given $a \in A$?

Assume that for any element x, you know whether $x \in B$ or not

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is “less complicated” than B.
Indeed we have …

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \ \text{ for some } f \text{ continuous}$$

Does a given $a \in A$?

Assume that for any element x, you know whether $x \in B$ or not

$$a \xmapsto{\quad f \quad} f(a)$$

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is "less complicated" than B.
Indeed we have …

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \;\text{ for some } f \text{ continuous}$$

Does a given $a \in A$?

Assume that for any element x, you know whether $x \in B$ or not

$$a \longmapsto f \longrightarrow f(a)$$

If $f(a) \in B$, then $a \in A$

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is “less complicated” than B.
Indeed we have …

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \ \text{ for some } f \text{ continuous}$$

Does a given $a \in A$?

Assume that for any element x, you know whether $x \in B$ or not

$$a \xmapsto{\quad f \quad} f(a)$$

If $f(a) \in B$, then $a \in A$

If $f(a) \notin B$, then $a \notin A$

In other terms, the belonging problem in A reduces via f to the belonging problem in B.

$A \leq_W B$ means intuitively than A is “less complicated” than B. Indeed we have …

$$A \leq_W B \quad \text{iff} \quad x \in A \Leftrightarrow f(x) \in B \;\text{ for some } f \text{ continuous}$$

Does a given $a \in A$?

Assume that for any element x, you know whether $x \in B$ or not

$$a \xmapsto{\quad f \quad} f(a)$$

If $f(a) \in B$, then $a \in A$

If $f(a) \notin B$, then $a \notin A$

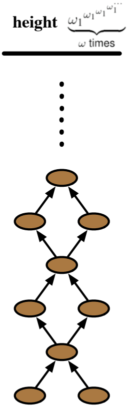
In other terms, the belonging problem in A reduces via f to the belonging problem in B.

Definition

- The collection of all subsets of $\mathcal{C}$ (i.e. all ω-languages) ordered by the relation $\leq_W$ is called *the Wadge hierarchy*.
- The collection of all Borel subsets of $\mathcal{C}$ ordered by the relation $\leq_W$ is called *the Borel Wadge hierarchy*.

Theorem

The Wadge hierarchy of Borel subsets of finite ranks has width 2 and height $\underbrace{\omega_1^{\omega_1^{\omega_1^{\omega_1^{\cdot^{\cdot^{\cdot}}}}}}}_{\omega \text{ times}}$ *(huge!).*



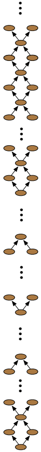
The Borel Hierarchy of finite ranks

height ω

$$
\begin{array}{ccccc}
 & & \vdots & & \\
\Sigma^0_3 & & & & \Pi^0_3 \\
 & \supsetneq & & \subsetneq & \\
 & & \Delta^0_3 & & \\
 & \subsetneq & & \supsetneq & \\
\Sigma^0_2 & & & & \Pi^0_2 \\
 & \supsetneq & & \subsetneq & \\
 & & \Delta^0_2 & & \\
 & \subsetneq & & \supsetneq & \\
\Sigma^0_1 & & & & \Pi^0_1 \\
 & \supsetneq & & \subsetneq & \\
 & & \Delta^0_1 & &
\end{array}
$$

The Wadge Hierarchy

height $\underbrace{\omega_1^{\omega_1^{\omega_1^{\omega_1^{\cdots}}}}}_{\omega \text{ times}}$



The Borel Hierarchy of finite ranks

height ω

$\vdots$

Σ^0_3 Π^0_3

Δ^0_3

Σ^0_2 Π^0_2

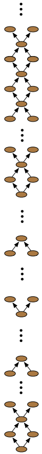
Δ^0_2

Σ^0_1 Π^0_1

Δ^0_1

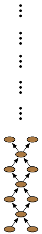
The Wadge Hierarchy

height $\underbrace{\omega_1^{\omega_1^{\omega_1^{\omega_1^{\cdots}}}}}_{\omega \text{ times}}$



The Wagner Hierarchy

height ω^ω



- The Wadge hierarchy is a very refined classification of ω-languages.
- The Wadge hierarchy is a very refined classification of "anything that can be coded as ω-languages".
- The Wadge hierarchy of more complex languages (like tree languages) also exists.
- Descriptive set theory provides a general framework for classification problems.

- The Wadge hierarchy is a very refined classification of ω-languages.
- The Wadge hierarchy is a very refined classification of "anything that can be coded as ω-languages".
- The Wadge hierarchy of more complex languages (like tree languages) also exists.
- Descriptive set theory provides a general framework for classification problems.

- The Wadge hierarchy is a very refined classification of ω-languages.
- The Wadge hierarchy is a very refined classification of "anything that can be coded as ω-languages".
- The Wadge hierarchy of more complex languages (like tree languages) also exists.
- Descriptive set theory provides a general framework for classification problems.

- The Wadge hierarchy is a very refined classification of ω-languages.
- The Wadge hierarchy is a very refined classification of "anything that can be coded as ω-languages".
- The Wadge hierarchy of more complex languages (like tree languages) also exists.
- Descriptive set theory provides a general framework for classification problems.