An introduction to descriptive complexity

IHPST 2024

November 15, 2024

Damiano Mazza, & Baptiste Chanus

Descriptive complexity : languages and computations

Logical description

- Finite structures over finite signatures
- Logical ressources for expressivity (higher order quantifiers, operators)

Decision algorithm

- Models of computations (Turing machines, circuits)
- Computational ressources (Time, space)

Descriptive complexity : languages and computations

Logical description

- Finite structures over finite signatures
- Logical ressources for expressivity (higher order quantifiers, operators)

Decision algorithm

- Models of computations (Turing machines, circuits)
- Computational ressources (Time, space)

Descriptive complexity

$$x \models F \iff x \in L$$

Descriptive complexity : languages and computations

Logical description

- Finite structures over finite signatures
- Logical ressources for expressivity (higher order quantifiers, operators)

Decision algorithm

- Models of computations (Turing machines, circuits)
- Computational ressources (Time, space)

Descriptive complexity

Input

$$x \models F \iff x \in L$$

Logical description

Computational description

Complexity Class

Definition

A **problem** L (on *string*) is a set of strings on an alphabet Σ.
We write $L \subseteq \Sigma$.

Definition

A **complexity class** is a set of problems that are solvable in *bounded ressources* (time, space...) in a given model of computations (Turing Machines).

Our approach what's different ?

~~Logical description~~ Boolean theories

- Finite models of sorted first order finite theories (adding axioms to signatures)
- Logical ressources ~~(higher order quantifiers, operators)~~ sorts and relations

Boolean Theories

Definition

A **Boolean theory** $\mathbb{T}$ is a triple

$$(\mathrm{Sort}(\mathbb{T}), \mathrm{Rel}(\mathbb{T}), \mathrm{Ax}(\mathbb{T}))$$

A Boolean theory $\mathbb{T}$ is **finite** if $\mathrm{Sort}(\mathbb{T})$, $\mathrm{Rel}(\mathbb{T})$ and $\mathrm{Ax}(\mathbb{T})$ are all finite.

Example : $\mathbb{S}\mathrm{tr}$

Definition

$$\begin{aligned}\mathrm{Sort}(\mathbb{S}\mathrm{tr}) &= \{N\}\\ \mathrm{Rel}(\mathbb{S}\mathrm{tr}) &= \{\leq\ \rightarrowtail N \times N, \mathrm{isOne} \rightarrowtail N\}\\ \mathrm{Ax}(\mathbb{S}\mathrm{tr}) &= \{\text{“} \leq \text{is a total order''}\}\end{aligned}$$

$$\begin{array}{ccccccccc} (N, \leq) & : & \mathsf{a} & \leq & \mathsf{b} & \leq & \mathsf{c} & \leq & \mathsf{d} \\ & & \uparrow & & \uparrow & & \uparrow & & \uparrow \\ \mathrm{isOne} & : & 0 & & 1 & & 0 & & 1 \end{array}$$

Other example : $\mathbb{G}\mathrm{rph}$

Definition

$$\mathrm{Sort}(\mathbb{G}\mathrm{rph}) = \{V\}$$
$$\mathrm{Rel}(\mathbb{G}\mathrm{rph}) = \{E \rightarrowtail V \times V\}$$
$$\mathrm{Ax}(\mathbb{G}\mathrm{rph}) = \emptyset$$

Extension of a theory

Definition

$\mathbb{T}$ extends $\mathbb{T}'$ iff :

- $\mathrm{Sort}(\mathbb{T}') \subseteq \mathrm{Sort}(\mathbb{T})$
- $\mathrm{Rel}(\mathbb{T}') \subseteq \mathrm{Rel}(\mathbb{T})$
- $\mathrm{Ax}(\mathbb{T}') \subseteq \mathrm{Ax}(\mathbb{T})$

Definition

$\mathbb{T}$ is a relational extension of $\mathbb{T}'$ iff :

- $\mathbb{T}'$ is an extension of $\mathbb{T}$
- $\mathrm{Sort}(\mathbb{T}') = \mathrm{Sort}(\mathbb{T})$

Extension of a theory

Definition

$\mathbb{T}$ extends $\mathbb{T}'$ iff :

- $\mathrm{Sort}(\mathbb{T}') \subseteq \mathrm{Sort}(\mathbb{T})$
- $\mathrm{Rel}(\mathbb{T}') \subseteq \mathrm{Rel}(\mathbb{T})$
- $\mathrm{Ax}(\mathbb{T}') \subseteq \mathrm{Ax}(\mathbb{T})$

Definition

$\mathbb{T}$ is a relational extension of $\mathbb{T}'$ iff :

- $\mathbb{T}'$ is an extension of $\mathbb{T}$
- $\mathrm{Sort}(\mathbb{T}') = \mathrm{Sort}(\mathbb{T})$

Note that there is a natural notion of projection from the extension to the base theory. (i.e. the one that forgets the extra information)

Generic form

Theorem (Generic Form)

Complexity class $\mathcal{C}$ is equal to set of extensions $\mathbb{T}$ of $\mathbb{S}\mathrm{tr}$.

Generic form

Theorem (Generic Form)

Complexity class $\mathcal{C}$ is equal to set of extensions $\mathbb{T}$ of $\mathbb{S}\mathrm{tr}$.

What does this mean ?

Generic form

Theorem (Generic Form)

Complexity class $\mathcal{C}$ is equal to set of extensions $\mathbb{T}$ of $\mathbb{S}\mathrm{tr}$.

Logic

For every problem $p \in \mathcal{C}$ and for every x $\in p$, we have at leat one model of $\mathbb{T}$ that projects onto x.

Computation

For every model m of $\mathbb{T}$ there is a problem p such that x $\in p$ if and only m projects onto x.

Fagin's Theorem (our version)

Theorem (Fagin (Boolean sauce))

NP is equal to the relational extensions of $\mathbb{S}\mathrm{tr}$.

Sketch of the proof

Logical description

Given a NP Turing machine :

- Give a theory such that all its finite models can project to accepting runs of the machine
- Is this a relational extension of $\mathbb{S}\mathrm{tr}$? (without detail)

Decision algorithm

Given a relational extension of $\mathbb{S}\mathrm{tr}$

- Give a Turing machine whose accepting runs are models of the theory
- Is this a NP Turing machine?

Logical encoding

Extending strings with table of symbols

$$\mathbb{S}\mathrm{tr} + \textbf{S,T} + \mathrm{Symb}_0, \mathrm{Symb}_1, \mathrm{Symb}_\Box \rightarrowtail T \times S + (\mathrm{State}_q) \rightarrowtail T$$

Axioms :

- **S, T** are finite chains (equipped with successors and max)
- $\mathrm{Symb}_{\{0,1,\Box\}}$ form a function from $T \times S$ to $\{0, 1, \Box\}$ and $(State_q)$ from T to Q
- State q_0 and blank symbols $\Box$ on work tape at time 0. State $accept$ at final state

i_0	i_1	i_2		i_{n-1}	i_n
0	0	1	...	0	0

	s_1	s_2	...	s_{n-1}	s_n	
t_1	1	1	...	0	0	q_0
t_2	0	1	...	0	1	q
$\vdots$			...			$\vdots$
t_{m-1}	1	0	...	0	0	q'
t_m	1	1	...	0	0	$accept$

Adding heads

$\mathbb{S}\mathrm{tr} + \textbf{S,T} + \mathrm{Symb}_{\{0,1,\square\}}, (\mathrm{State}_q) + \mathrm{wHead} \rightarrowtail T \times S + \mathrm{iHead} \rightarrowtail T \times N$

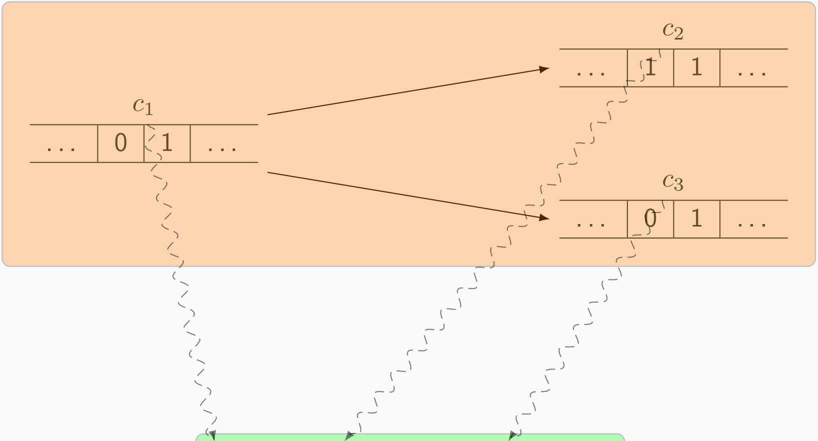
Axioms :

- wHead (resp. iHead) are functions from T to S (resp. N)
- wHead and iHead don't move more than one case
- The work tape is unchanged at positions where the head is not found

i_0	i_1	i_2		i_{n-1}	i_n
0	0	1	...	0	0

	s_1	s_2	...	s_{n-1}	s_n	
t_1	1	1	...	0	0	q_0
t_2	0	1	...	0	0	q
$\vdots$	$\vdots$	$\vdots$	...	$\vdots$	$\vdots$	$\vdots$
t_{m-1}	1	0	...	0	0	q'
t_m	1	1	...	0	1	$accept$

Axioms for the transitions



$$\Phi_{c_1}(t) \;\Rightarrow\; \Phi_{c_2}(t+1) \;\vee\; \Phi_{c_3}(t+1)$$

Polynomiality

All Turing mahines are represented !

But $\#T$ is exactly the time of the Turing machine

Model checking

done fast

Computing a model of a theory

Given a relational extension of $\mathbb{S}\mathrm{tr}$ adding $R_1, \ldots, R_p$ and axioms, we give a machine M such that its accepting runs are exactly the models of the extension :