Descriptive Complexity: Trakhtenbrot's Theorem, SO Logic and Fagin's Theorem

Danai Balla

May 19, 2020

ALMA
INTER-INSTITUTIONAL GRADUATE PROGRAM
"ALGORITHMS, LOGIC AND DISCRETE
MATHEMATICS"

Overview

We will prove two theorems:

- Trakhtenbrot's theorem: The set of finitely satisfiable sentences is not recursive.
  Corollary: The set of finitely valid sentences is not recursively enumerable.
- Fagin's theorem: $\exists$SO captures NP ($\exists$SO=NP)

Contents

1 Trakhtenbrot's Theorem

2 Second Order Logic and Fagin's Theorem

- Second Order Logic
- Fagin's Theorem
- Results of Fagin's Theorem

Introduction

Completeness theorem for FO:

A sentence Φ is valid iff it is provable in some formal system.

This implies that the set of all valid FO sentences is recursively enumerable: We can have a TM try all possible proofs in lexicographic order and report "yes" if one of them is a proof of the given expression.

We will show that this completeness fails when only finite models are allowed.

Trakhtenbrot's theorem

Definition

Given a vocabulary σ, a sentence Φ in that vocabulary is called *finitely satisfiable* if there is a finite structure $\mathcal{A} \in \mathrm{STRUCT}[\sigma]$ such that $\mathcal{A} \models \Phi$

The sentence Φ is called *finitely valid* if $\mathcal{A} \models \Phi$ for all finite structures $\mathcal{A} \in \mathrm{STRUCT}[\sigma]$.

Theorem (Trakhtenbrot)

For every relational vocabulary σ with at least one binary relation symbol, it is undecidable whether a sentence Φ of vocabularty σ is finitely satisfiable.

Proof

Proof idea

For every Turing Machine M we construct a sentence Φ_M of vocabulary σ such that Φ_M is finitely satisfiable iff M halts on the empty input. The latter is well known to be undecidable.

Let $M = (Q, \Sigma, \Delta, \delta, q_0, Q_a, Q_r)$ be a deterministic Turing machine with a one way infinite tape.

We can assume wlog that $\Delta = \{0, 1\}$ where 0 represents the blank symbol.

Proof

We define σ so that its structures represent computations of M

$$\sigma = \{<, \underline{\text{min}}, T_0(\cdot,\cdot), T_1(\cdot,\cdot)\,(H_q(\cdot,\cdot))_{q\in Q}\}$$

Where

- $<$: Linear order and $\underline{\text{min}}$ constant symbol for the minimal element with respect to $<$
- T_0, T_1: Tape predicates
  $T_i(p,t)$ indicates that position p at time t contains i, for $i = 0, 1$
- H_q: Head predicates
  $H_q(p,t)$ indicates that at time t, the machine is in state q and its head is in position p

Proof

We define Φ_M to be the conjunction of the following sentences

- A sentence stating that $<$ is a linear ordering and $\underline{\text{min}}$ is its minimal element.
- A sentence defining the initial configurations of M:
  $H_{q_0}(\underline{\text{min}}, \underline{\text{min}}) \wedge \forall p T_0(p, \underline{\text{min}})$
- A sentence stating that in every configuration of M, each cell of the tape contains exactly on element of Δ:
  $\forall p \forall t (T_0(p, t) \leftrightarrow (\neg T_1(p, t))$

Proof

- A sentence stating that at any time the machine is exactly in one state:
$$\forall t \exists ! p \left(\bigvee_{q \in Q} H_q(p,t) \right) \land \neg \exists p \exists t \left(\bigvee_{q,q' \in Q, q \neq q'} H_q(p,t) \land H_{q'}(p,t) \right)$$
- A set of sentences stating that T_i's and H_q's respect the transitions of M.
  For example if $\delta(q,0) = (q',1,l)$, this transition is represented by the conjunction of
$$\forall p \forall t \begin{pmatrix} p \neq \underline{\min} \\ \land T_0(p,t) \\ \land H_q(p,t) \end{pmatrix} \rightarrow \begin{pmatrix} T_1(p,t+1) \\ \land H_{q'}(p-1,t+1) \\ \land \forall p'(p \neq p' \rightarrow (\bigwedge_{i=0,1} T_i(p',t+1) \leftrightarrow T_i(p,t)) \end{pmatrix}$$

Proof

and

$$\forall p \forall t \begin{pmatrix} p = \underline{min} \\ \wedge T_0(p,t) \\ \wedge H_q(p,t) \end{pmatrix} \rightarrow \begin{pmatrix} T_1(p, t+1) \\ \wedge H_{q'}(p, t+1) \\ \wedge \forall p'(p \neq p' \rightarrow (\bigwedge_{i=0,1} T_i(p', t+1) \leftrightarrow T_i(p', t)) \end{pmatrix}$$

- Finally a sentence stating that at some point M is in halting state:
  $\exists p \exists t \bigvee_{q \in Q_a \cup Q_r} H_q(p,t)$

M halts on the empty input iff Φ_M has a finite model.
Since testing if M halts on the empty input is undecidable, then so is finite satisfiability for Φ_M

Corollary

Corollary

For any vocabulary containing at least one binary relation symbol, the set of finitely valid sentences is not recursively enumerable.

Proof.

The set of finitely satisfiable sentences is recursively enumerable: We can enumerate all pairs $(\mathcal{A}, \Phi)$ where $\mathcal{A}$ is finite and output Φ whenever $\mathcal{A} \models \Phi$. Assume that the set of finitely valid sentences is r.e.,then since Φ is valid iff $\neg\Phi$ is not finitely satisfiable, we conclude that the set of finitely satisfiable sentences is recrusive, which contradicts Trakhtenbrot's theorem. □

Contents

1 Trakhtenbrot's Theorem

2 Second Order Logic and Fagin's Theorem

- Second Order Logic
- Fagin's Theorem
- Results of Fagin's Theorem

Table of Contents

1 Trakhtenbrot's Theorem

2 Second Order Logic and Fagin's Theorem

- Second Order Logic
- Fagin's Theorem
- Results of Fagin's Theorem

Second Order Logic-Definition

Definition (Second Order Logic)

We assume that for every $k > 0$ there are infinitely many variables $X_1^k, X_2^k, \dots$ ranging over k-ary relations.
Given a vocabulary σ that consists of relation and constant symbols, we define:

- **Terms:** FO variables and constant symbols. x is the only free variable of a term x and constant c has no free variables
- **Atomic formulae:**
  - FO atomic formulae
  - $X(t_1, \dots, t_k)$ where $t_1, \dots, t_k$ are terms and X is a SO variable of arity k. (t_i's: free FO variables, X: free SO variable)
- $\wedge, \vee, \neg$, quantification as in FO
- If $\phi(\vec{x}, Y, \vec{X})$ is a formula, then $\forall Y \phi(\vec{x}, Y, \vec{X})$ and $\exists \phi(\vec{x}, Y, \vec{X})$ are formulae whose free variables are $\vec{x}$ (FO) and $\vec{X}$ (SO).

Semantics of SO

Definition (Semantics of SO logic)

For each formula $\phi(\vec{x}, \vec{X})$ we define the notion of $\mathcal{A} \models \phi(\vec{b}, \vec{B})$ where $\vec{b}$ is a tuple of elements of A with the same length as $\vec{x}$ and if $\vec{x} = (X_1, \ldots, X_l)$, $\vec{B} = (B_1, \ldots, B_l)$ with $B_i \subseteq A^{\text{arity}(X_i)}$.

For constructions that are different from those of FO:

- If $\phi(\vec{x}, \vec{X})$ is $X(t_1, \ldots, t_k)$, t_i terms with free variables among $\vec{x}$, then $\mathcal{A} \models \phi(\vec{b}, \vec{B})$ iff $(t_1^{\mathcal{A}}(\vec{b}), \ldots, t_k^{\mathcal{A}}(\vec{b}))$ is in $\vec{B}$.
- If $\phi(\vec{x}, \vec{X})$ is $\exists Y \psi(\vec{x}, Y, \vec{X})$ with Y k-ary then $\mathcal{A} \models \phi(\vec{b}, \vec{B})$ if for some $C \subseteq A^k$ we have $\mathcal{A} \models \psi(\vec{b}, C, \vec{B})$
- If $\phi(\vec{x}, \vec{X})$ is $\forall Y \psi(\vec{x}, Y, \vec{X})$ with Y k-ary then $\mathcal{A} \models \phi(\vec{b}, \vec{B})$ if for all $C \subseteq A^k$ we have $\mathcal{A} \models \psi(\vec{b}, C, \vec{B})$

Existential and Universal SO Logic

Definition

Existential SO *logic* or ∃SO is defined as the restriction of SO that consists of the formulae of the form

$$\exists X_1 \dots \exists X_n \phi$$

where ϕ does not have any second order quantification.
If the second order quantifier prefix consists only of universal quantifiers, we speak of the *universal* SO *logic* or ∀SO

Examples in ∃SO

Example

$\Phi_{\text{3-color}} \equiv (\exists R^1 \exists Y^1 \exists B^1 \forall x((R(x) \vee Y(x) \vee B(x)) \wedge (\forall y(E(x,y) \rightarrow \neg(R(x) \wedge R(y)) \wedge \neg(Y(x) \wedge Y(y)) \wedge \neg(B(x) \wedge B(y)))$

A graph G satisfies $\Phi_{\text{3-color}}$ iff G is 3-colorable.

Example

$\Phi_{\text{SAT}} \equiv (\exists S)(\forall x)(\exists y)((P(x,y) \wedge S(y)) \vee (N(x,y) \wedge \neg S(y)))$

Table of Contents

1 Trakhtenbrot's Theorem

2 Second Order Logic and Fagin's Theorem

- Second Order Logic
- Fagin's Theorem
- Results of Fagin's Theorem

A general definition

Definition

Let $\mathcal{K}$ be a complexity class, $\mathcal{L}$ a logic and $\mathcal{C}$ a class of finite structures. We say that $\mathcal{L}$ *captures* $\mathcal{K}$ *on* $\mathcal{C}$ if the following hold:

1. The data complexity of $\mathcal{L}$ on $\mathcal{C}$ is $\mathcal{K}$; that is, for every $\mathcal{L}$-sentence Φ, testing if $\mathcal{A} \models \Phi$ is in $\mathcal{K}$, provided $\mathcal{A} \in \mathcal{C}$.
2. For every property $\mathcal{P}$ of structures from $\mathcal{C}$ that can be tested with complexity $\mathcal{K}$, there is a sentence $\Phi_{\mathcal{P}}$ of $\mathcal{L}$ such that $\mathcal{A} \models \Phi_{\mathcal{P}}$ iff $\mathcal{A}$ has the property $\mathcal{P}$, for every $\mathcal{A} \in \mathcal{C}$.

If $\mathcal{C}$ is the class of all finite structures, we say that $\mathcal{L}$ captures $\mathcal{K}$.

Fagin's Theorem

Theorem (Fagin)

$\exists$SO *captures* NP.

Proof:

Every $\exists$SO sentence Φ can be evaluated in NP: Suppose Φ is $\exists S_1 \cdots \exists S_n \phi$ where ϕ is FO. Given $\mathcal{A}$, the NTM guesses $S_1, \ldots, S_n$ and checks if $\phi(S_1, \ldots, S_n)$ holds. The latter can be done in polynomial time in $||\mathcal{A}||$ plus the size of $S_1, \ldots, S_n$ and thus in polynomial time in $||\mathcal{A}||$.

Proof

Next, we show that every NP-property of finite structures can be expressed in ∃SO.

Suppose we are given a property $\mathcal{P}$ of σ-structures that can be tested on encodings of σ-structures by a nondeterministic polynomial time TM $M = (Q, \Sigma, \Delta, \delta, q_0, Q_a, Q_r)$ with a one way infinite tape that runs in time n^k (and $Q = \{q_0, \ldots, q_{m-1}\}$)

We assume wlog that M visits the entire input, that $\Sigma = \{0, 1\}$ and Δ extends Σ with the blank symbol "$-$".

The sentence describing acceptance by M on encodings of structures from STRUCT[σ] will be of the form

$$\exists L \exists T_0 \exists T_1 \exists T_2 \exists H_{q_0} \cdots \exists H_{q_{m-1}} \Psi$$

Proof

Ψ is a sentence of vocabulary $\sigma \cup \{T_0, T_1, T_2\} \cup \{H_q | q \in Q\}$, L is binary, other symbols are of arity $2k$, and

- L is a linear order of the universe

M runs in time n^k and visits at most n^k cells so we can model positions on the tape and time as k-tuples $\vec{p}, \vec{t}$

- T_0, T_1, T_2: Tape predicates
  $T_i(\vec{p}, \vec{t})$ indicates that position $\vec{p}$ at time $\vec{t}$ contains i, for $i = 0, 1$ and for $i = 2$ contains the blank symbol
- H_q: Head predicates
  $H_q(\vec{p}, \vec{t})$ indicates that at time $\vec{t}$ the machine is in state q and its head is in position $\vec{p}$.

Proof

We define Ψ as the conjunction of the following sentences:

- The sentence stating that L defines a linear ordering
- The sentence stating that
  - In every configuration of M each cell of the tape contains exactly one element of Δ
  - At any time the machine is in exactly one state
  - At some time M enters a state from Q_a

(Same as in the proof of Trakhtenbrot's theorem)

Proof

- Sentences stating that T_i's and H_q's respect the transitions of M:
  For every $a \in \Delta$ and for every $q \in Q$ we have a sentence

$$\bigvee_{(q',b,move)\in\delta(q,a)} \alpha_{(q,a,q',b,move)}$$

  where $move \in \{l, r\}$ and $\alpha_{(q,a,q',b,move)}$ describing the transition in which upon reading a in state q the machine writes b, makes move $move$ and enters state q' (written same as in Trakhtenbrot's theorem).

Proof

- The sentence stating that at time 0 the tape contains the encoding of the structure followed by blanks:

  Suppose we have formulae $\iota(\vec{p})$ and $\xi(\vec{p})$ of vocabulary $\sigma \cup L$ such that $\mathcal{A} \models \iota(\vec{p})$ iff the $\vec{p}$-th position of $enc(\mathcal{A})$ is 1 and $\mathcal{A} \models \xi(\vec{p})$ iff $\vec{p}$ exceeds the length of $enc(\mathcal{A})$ (will be defined in a bit).

  Then the sentence is

$$\forall\vec{p}\forall\vec{t}\left(\neg\exists\vec{u}(\vec{u} <_k \vec{t}) \rightarrow \begin{bmatrix} (\iota(\vec{p}) \leftrightarrow T_1(\vec{t},\vec{p})) \\ \wedge(\xi(\vec{p}) \leftrightarrow T_2(\vec{t},\vec{p})) \end{bmatrix}\right)$$

Proof

For the case of $\sigma = \{E\}$ with E binary (to simplify the notation):

Assume that the universe is $\{0, \dots, n-1\}$ where $(i, j) \in L$ iff $i < j$.

The graph is encoded by the string $0^n 1 \cdot s$ where s is a string of length n^2 s.t it has 1 in position $un + v$ for $0 \leq u, v \leq n-1$ iff $(u, v) \in E$ and $\vec{p}$ represents the position $p_1 n^{k-1} + \cdots + p_{k-1} n + p_k$

Then $\iota(\vec{p})$ is equivalent to the disjunction of $\sum_{i=1}^{k} p_i n^{k-1} = n$ and

$$\exists u \leq (n-1) \exists v \leq (n-1) \left((n+1) + un + v = \sum_{i=1}^{k} p_i n^{k-1} \wedge E(u, v) \right)$$

$\xi(\vec{p})$ says that $\vec{p}$, considered as a number, exceeds the length of $enc(\mathcal{A})$.

Table of Contents

1 Trakhtenbrot's Theorem

2 Second Order Logic and Fagin's Theorem

- Second Order Logic
- Fagin's Theorem
- Results of Fagin's Theorem

Corollary

Corollary (Cook)

SAT is NP-complete

Proof.

Let $\mathcal{P}$ be a problem (a class of σ-structures) in NP. By Fagin's theorem, there is an $\exists$SO sentence $\Phi \equiv \exists S_1 \ldots \exists S_n \phi$ s.t. $\mathcal{A}$ is in $\mathcal{P}$ iff $\mathcal{A} \models \Phi$. Let $X = \{S_i(\vec{a}) \mid i = 1, \ldots, n, \vec{a} \in A^{\text{arity}(S_i)}\}$. We construct a propositional formula $\alpha_\phi^\mathcal{A}$ from ϕ by:

- Replacing each $\exists x \psi(x, \cdot)$ by $\bigvee_{a \in A} \psi(a, \cdot)$
- Replacing each $\forall x \psi(x, \cdot)$ by $\bigwedge_{a \in A} \psi(a, \cdot)$
- Replacing each $R(\vec{a})$ for $R \in \sigma$ by its truth value in $\mathcal{A}$

In $\alpha_\phi^\mathcal{A}$ the variables are of the form $S_i(\vec{a})$ (they come from X).

$\mathcal{A} \models \Phi$ iff $\alpha_\phi^\mathcal{A}$ is satisfiable and $\alpha_\phi^\mathcal{A}$ can be constructed by a deterministic logspace machine. □

Proposition

3-SAT *is* NP*-complete via first order reductions.*

Proof.

Let $\mathcal{A} \in \text{STRUCT}[\langle P^2, n^2 \rangle]$ be an instance of SAT with $n = ||\mathcal{A}||$. Each clause c of $\mathcal{A}$ is replaced by $2n$ clauses as follows:

$$([x_1]^c \vee d_1) \wedge (\overline{d_1} \vee [x_2]^c \vee d_2) \wedge (\overline{d_2} \vee [x_3]^c \vee d_3) \wedge \cdots$$
$$\wedge (\overline{d_n} \vee [\overline{x_1}]^c \vee d_{n+1}) \wedge (\overline{d_{n+1}} \vee [\overline{x_2}]^c \vee d_{n+2}) \wedge \cdots \wedge (\overline{d_{2n-1}} \vee [\overline{x_n}]^c) = c'$$

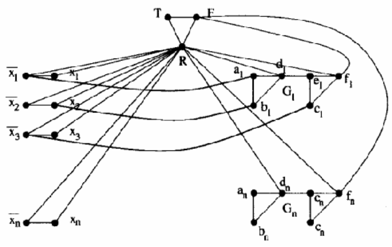
Where x_i's are the instance literals, d_i's are new variables and $[l]^c$ means the literal l if it occurs in c and **false** otherwise.

c is satisfiable iff c' is satisfiable and c' is definable in a first order way from c. □

Proposition

3-color *is* NP*-complete via first order reductions*

Proof.



We will show that 3-SAT$\leq_{fo}$3-COLOR. $\mathcal{A}$ instance of 3-SAT, $||\mathcal{A}|| = n$. We construct graph $f(\mathcal{A})$ s.t. $f(\mathcal{A})$ 3-colorable iff $\mathcal{A} \in$3-SAT (see figure)
In the figure,
G_1 encodes clause $\overline{x}_1 \vee x_2 \vee \overline{x}_3$
□

Separating Complexity Classes

Since coNP consists of the problems whose complements are in NP, and the negation of an ∃SO sentence is an ∀SO sentence, we obtain

Corollary

∀SO *captures* coNP

If we have two complexity classes $\mathcal{K}_1$ and $\mathcal{K}_2$ captured by logics $\mathcal{L}_1$ and $\mathcal{L}_2$, we could prove that $\mathcal{K}_1 \neq \mathcal{K}_2$ by showing that some problem definable in $\mathcal{L}_1$ is inexpressible in $\mathcal{L}_2$ or vice versa.

Separating ∀SO from ∃SO (over finite structures) would resolve the "PTIME vs NP" problem:

$$\forall \text{SO} \neq \exists \text{SO} \Rightarrow \text{NP} \neq \text{coNP} \Rightarrow \text{PTIME} \neq \text{NP}$$

Fagin's Theorem and the Polynomial Hierarchy (1/3)

- Levels of PH: Σ_1^P =NP, $\Sigma_{k+1}^p = \mathrm{NP}^{\Sigma_k^p}$ and Π_k^p the set of complements of problems from Σ_k^p
- Σ_k^1 the class of SO sentences of the form

$$(\exists \cdots \exists)(\forall \cdots \forall)(\exists \cdots \exists) \cdots \phi$$

with k quantifier blocks.
Π_k^1 defined the same way but the first block of quantifiers is universal.

Fagin's Theorem and the Polynomial Hierarchy (2/3)

Corollary

For each $k \geq 1$
(a) Σ^1_k *captures* Σ^p_k *and*
(b) Π^1_k *captures* Π^p_k
In particular, SO captures the polynomial hierarchy.

Inductive argument for (a) (sketch)
Base: Fagin's theorem
Consider a problem in Σ^p_{k+1}. By Fagin's theorem, there exists an $\exists$SO sentence Φ corresponding to the NP machine with additional pedicates expressing Σ^p_k properties and these properties are expressed by hypothesis by a Σ^k_1 formula. We push the second order quantifiers outwards and we have a Σ^{k+1}_1 formula.

Fagin's Theorem and the Polynomial Hierarchy (3/3)

The extra quantifier alternation arises when the predicates for Σ_k^p are negated:
Suppose we have a formula $\exists \cdots \exists \phi(P)$ where P is expressed by $\exists \cdots \exists \psi$ with ψ FO and P may occure negatively. Then puttings the resulting formula in the prenex form we have a formula of the form $(\exists \cdots \exists)(\forall \cdots \forall)$.
For example $\exists \cdots \exists \neg(\exists \cdots \exists \psi)$ is equivalent to $\exists \cdots \exists \forall \cdots \forall \neg \psi$.

Thank you

:)

Bibliography

- Immerman, Neil. (1999) Descriptive Complexity. Springer-Verlag New York.
- Leonid Libkin, "Elements of Finite Model theory" [2012]