DIMACS Technical Report 97-54
September 1997

Special Year on Logic and Algorithms Tutorial Notes: Expressive Powers of Logics (Tutorial Lectures by Phokion Kolaitis)

by

Eric Allender[1]
(Editor, SYLA Tutorial Notes)
Dept. of Computer Science
Rutgers University
New Brunswick, New Jersey 08903

[1]Permanent Member

DIMACS is a partnership of Rutgers University, Princeton University, AT&T Labs, Bellcore, and Bell Labs.

DIMACS is an NSF Science and Technology Center, funded under contract STC–91–19999; and also receives support from the New Jersey Commission on Science and Technology.

ABSTRACT

The 1995-1996 DIMACS Special Year on Logic and Algorithms began with three week-long tutorial sessions on the topics of the special year:

- Finite Model Theory
- Proof Complexity
- Computer-Aided Verification

These notes on the lectures given by Phokion Kolaitis were compiled by Eric Allender from the notes supplied by Randall J. Pruim, Brenda J. Latka, Jim Rogers, Maria-Luisa Bonet, and Kousha Etessami.

Additional material from the tutorials may be found from the www page

`http://dimacs.rutgers.edu/Workshops/SYLA-Tutorials/program.html`

Session I: Monday, August 14

Track A: Expressive Power of Logics

Speaker: P. Kolaitis

Finite Model Theory (FMT) = study of logics on classes of finite structures

- logics: First Order (FO) + extensions
- extensions: 2nd order (SO), fragments of SO, fixpoint ops, infinitary logics

Reasons for developing FMT

- connection to and application to CS (database & complexity)
- study in own right
  - traditional math. logic focussed on either all structures or fixed structure of interest (e.g., $\mathbb{N} = (\omega, +, *)$)
  - new phenomena occur in the case where one considers only classes of finite structures (e.g., 0-1 laws)

Basics & Notation

- **vocabulary** = collection of relation symbols (R_i) with arities (n_i) (no functions or constants)
- **structure** $\mathfrak{A}$ = universe $|\mathfrak{A}|$ together with interpretations of the relation symbols on the universe $(R_i^{\mathfrak{A}})$
- **class of structures** = collection of structures closed under isomorphisms
- **boolean query** on a class of structures, $\mathcal{C}$, over a vocab Σ = function $Q : \mathcal{C} \to \{0,1\}$ such that
$$Q(\mathfrak{A}) = Q(\mathfrak{B}) \quad \text{whenever} \quad \mathfrak{A} \cong \mathfrak{B}$$
- ***k*-ary query** = $Q : \mathcal{C} \to \{k\text{-ary relations}\}$, such that
  - if $\mathfrak{A} \in \mathcal{C}$ then $Q(\mathfrak{A}) \subseteq \mathcal{P}(|\mathfrak{A}|^k)$
  - if $h : \mathfrak{A} \to \mathfrak{B}$ is an isomorphism, then $Q(\mathfrak{B}) = h(Q(\mathfrak{A}))$

Examples of queries on class of finite graphs

- transitive closure (of edge relation): $Q(G) = \{(a,b) : \exists \text{ path } a \to b\}$
- articulation domain: $Q(G) = \{a : a \text{ is articulation pt. of } G\}$
- 2-colorability: $Q(G) = 1$ if G 2-colorable, else 0

Definability of Queries

- boolean case: $\exists$ sentence ψ such that $Q(\mathfrak{A}) \Leftrightarrow \mathfrak{A} \models \psi$.
- k-ary case: $\exists$ formula φ such that $Q(\mathfrak{A}) = \{\vec{a} \in \mathcal{C} : \mathfrak{A} \models \varphi[\vec{x}/\vec{a}]\}$

Examples of logics

- **FO** = atomic formulas ($x_i = x_j, R(\vec{x})$) and closure under $\wedge$, $\vee$, $\neg$, and $(\exists x)\varphi(x)$, $(\forall x)\varphi(x)$, where quantifiers are over elements of the universe of the structure.
- **SO** = FO + formulas of the type $(\exists S)\varphi$, $(\forall S)\varphi$, where φ is first order and S is a relation symbol of some arity.
- **Monadic SO** = SO restricted so that relations in second order quantification (S above) have arity 1. This is quantification over subsets of the universe.
- **SO$\exists$** = SO where all second order quantifiers are existential.
- **Monadic SO$\exists$** = restriction of second order quantifiers in SO$\exists$ to arity 1. (I.e., all second order quantification is existential quantification over subsets of the universe of the structure.)
- SO$\forall$, monadic SO$\forall$ defined analogously

  note: SO$\exists$ also called Σ^1_1, SO$\forall$ also called Π^1_1

Properties of graphs and the logics in which they can be expressed

- FO: isolated node, existence of isolated nodes, "x has degree 3", "G is 3-regular"
- Monadic SO: 2-colorability

$$\begin{aligned}\exists B \exists R \qquad & [\forall x\,((B(x) \vee R(x)) \wedge (\neg B(x) \vee \neg R(x))) \\ & \wedge\, (\forall x \forall y E(x,y) \Rightarrow [R(x) \Leftrightarrow B(y)])]\end{aligned}$$

- well-order over **all** structures is Monadic SO$\forall$ definable but not FO definable. Proof uses compactness, hence does not imply same result over finite models.

Ehrenfeucht-Fraissé games (EF games): tool for showing non-expressibility

- set-up for an EF game:
  - 2 players: Spoiler and Duplicatior
  - 2 structures over same vocabulary ($\mathfrak{A}$ and $\mathfrak{B}$)
  - fixed number of rounds k

- **play** of the EF game: At each round, Spoiler selects one element from one of the structures, then Duplicator selects one element from the other structure. This determines elements $a_1, \ldots, a_k \in |\mathfrak{A}|$ and $b_1, \ldots b_k \in |\mathfrak{B}|$.
- **Duplicator wins the play** of the EF game if the mapping $a_i \mapsto b_i$ is a partial isomorphism from $\mathfrak{A}$ to $\mathfrak{B}$. (Partial isomorphism means for any phi, $\mathfrak{A} \models \varphi(x_1/a_1, \ldots, x_k/a_k) \Leftrightarrow \mathfrak{B} \models \varphi(x_1/b_1, \ldots, x_k/b_k)$.)
- **Duplicator wins the k-round EF Game** on structures $\mathfrak{A}$ and $\mathfrak{B}$ iff there is a **winning strategy** for Duplicator, that is a strategy by which Duplicator wins any play of the game, regardless of which elements are selected by Spoiler.

Notation. ($\mathfrak{A}$ and $\mathfrak{B}$ are structures over the same fixed vocabulary.)

- $qr(\varphi)$ = **quantifier rank** of φ = depth of nesting of quantifiers (*not* quantifier blocks) in the formula φ. (Easily defined inductively, atomics have qr 0, logical connections take max of components, quantifiers increase by 1.)
- $\mathfrak{A} \sim_k \mathfrak{B}$ iff Duplicator wins the k-round EF game on $\mathfrak{A}$ and $\mathfrak{B}$.
- $\mathfrak{A} \equiv_k \mathfrak{B}$ iff for all sentences φ in the fixed vocabulary of the structures, $qr(\varphi) \leq k \Rightarrow (\mathfrak{A} \models \phi \Leftrightarrow \mathfrak{B} \models \varphi)$

Theorem 1 EF Thm. *Fix $k \in \omega$, $\mathfrak{A}$, $\mathfrak{B}$ (structures over same vocabulary). Then TFAE*

1. $\mathfrak{A} \sim_k \mathfrak{B}$
2. $\mathfrak{A} \equiv_k \mathfrak{B}$

Moreover, the index of $\equiv_k$ is finite, and each of the equivalence classes is FO definable by a sentence of quantifier rank k.

note: only direction needed is (1) $\Rightarrow$ (2), which is straightforward to prove.

Corollary 1 *For any boolean query, Q, on a class of structures $\mathcal{C}$, TFAE*

1. *Q is FO definable on $\mathcal{C}$.*
2. *there is some k such that if $\mathfrak{A}, \mathfrak{B} \in \mathcal{C}$, $Q(\mathfrak{A}) = 1$, and $\mathfrak{A} \sim_k \mathfrak{B}$, then $Q(\mathfrak{B}) = 1$.*

note: (1) $\Rightarrow$ (2) uses (1) $\Rightarrow$ (2) of theorem. (2) $\Rightarrow$ (1) uses (2) $\Rightarrow$ (1) of theorem and the "moreover" part.

Corollary 2 *The following is a method for showing that a boolean query Q is not FO definable over a class $\mathcal{C}$: Show that for every $k \geq 1$ there exist $\mathfrak{A}_k$ and $\mathfrak{B}_k$ such that*

- *$\mathfrak{A}_k \in \mathcal{C}$ and $\mathfrak{B}_k \in \mathcal{C}$,*

- *$Q(\mathfrak{A}_k) = 1$ and $Q(\mathfrak{B}_k) = 0$,*
- *and $\mathfrak{A}_k \sim_k \mathfrak{B}_k$. (I.e., that Duplicator wins the k-round game.)*

Moreover, this method is complete in the sense that if Q is not FO definable over $\mathcal{C}$ then such models $\mathfrak{A}_k$, $\mathfrak{B}_k$ are guaranteed to exist, although constructing them and proving that they have these properties may be nontrivial.

Examples on finite graphs

- "Even number of nodes" is not FO definable on finite graphs.
- Eulerian graph not FO definable on finite graphs.
  - Eulerian cycle = cycle that visits all edges exactly once.
  - Eulerian graph = graph with Eulerian cycle = graph in which every node has even degree.
  - show result using graphs G and H with $2k+2$ and $2k+4$ nodes and the following edge relations (in both graphs):$E(n,m) \Leftrightarrow (n \in \{0,1\} \Leftrightarrow m \geq 2)$. G is Eulerian, H is not because of the degrees of nodes 0 and 1.

Implementation Difficulties

- how does one find the structures $\mathfrak{A}_k$ and $\mathfrak{B}_k$ for all k? [Ans. be creative]
- once candidate structures have been found, how does one prove rigourously that $\mathfrak{A} \sim_k \mathfrak{B}$?
  - for some classes, there is a complete analysis of the $\sim_k$ equivalence classes.
  - else, we can attempt to develop a library of winning strategies for the Duplicator. That is, we want general sufficient conditions under which the Duplicator wins.

Another example: Linear order on n elements (L_n)

- $L_6 \not\sim_3 L_7$: Spoiler chooses middle element of L_7 at first move.
- generally, $L_n \sim_k L_m \Leftrightarrow m = n$ or $(m \geq 2^k - 1 \;\&\; n \geq 2^k - 1)$.
- above result implies that "even size" is not FO definable on finite linear orders.

Session II: Tuesday, August 15
Track A: Expressive Power of Logics
Speaker: Kolaitis

Summary of Session I

Theorem 2 *TFAE*

1. $\mathfrak{A} \sim_r \mathfrak{B}$ *Duplicator wins r move EF game.*
2. $\mathfrak{A} \equiv_r \mathfrak{B}$ $\mathfrak{A}, \mathfrak{B}$ *satisfy same FO sentences of quantifier rank r.*

The Method

Given $\mathcal{C}$ a class of structures, Q a Boolean query on $\mathcal{C}$, to show that Q is not FO-definable in $\mathcal{C}$, it is sufficient to show that for every $r \geq 1$ there are $\mathfrak{A}_r, \mathfrak{B}_r$ such that

- $\mathfrak{A}_r \in \mathcal{C}$, $\mathfrak{B}_r \in \mathcal{C}$
- $Q(\mathfrak{A}_r) = 1$, $Q(\mathfrak{B}_r) = 0$
- $\mathfrak{A} \sim_r \mathfrak{B}$, that is, the Duplicator wins.

This method is both sound and complete.

Fagin, Stockmeyer, Vardi (1993)
Arora, Fagin (1994)
Schewentick (1994)

A Library of Winning Strategies

We define the connection graph of $\mathfrak{A} = (A, R_1, \ldots, R_m)$ by the edge relation $E_{\mathfrak{A}}(a, b) \Leftrightarrow \exists i\ R_i(\ldots a \ldots b \ldots) \vee\ R_i(\ldots b \ldots a \ldots)$. This is also called the Gaifman graph. It is an undirected graph.

We define $N(a, d)$ to be a and all points in A whose distance from a in $E_{\mathfrak{A}}$ is less than d, $d \geq 1$.

We define the d-type of a as the isomorphism type of $N(a, d)$ with a as a distinguished node. Note: this is using the induced substructure from A. $N(a, d) \cong N(b, d)$ if there is an isomorphism $h : N(a, d) \rightarrow N(b, d)$ and $h(a) = b$.

Examples

- L_n, the linear order with n elements. $N(a, d) = L_n$ if $d \geq 2$
- $\overline{K_n}$, the graph on n nodes with no edges. $N(a, d) = \{a\}$ for all d.
- C_n, the cycle on n vertices. If $n \geq 2d$, $N(a, d)$ is a path with $2d - 1$ nodes and a as the middle point.

Definition 1 *$\mathfrak{A}, \mathfrak{B}$ are d-equivalent if for every d-type they have the same number of points of that d-type.*

Theorem 3 (essentially Monf 1961) *For every $r \geq 1$ there is some $d \geq 1$ such that if $\mathfrak{A}, \mathfrak{B}$ are d-equivalent then $\mathfrak{A} \sim_r \mathfrak{B}$.*

In fact, any $d \geq 3^r$ works. So this equivalence relation is coarser.

Hint of proof:

Show that Duplicator can win r-move EF game by maintaining the j-matching condition:

if $a_1, \ldots, a_j, b_1, \ldots, b_j$ are the points played so far, then

$$\mathfrak{A}|(\bigcup_{i \leq j} N(a_i, 3^{r-j})) \cong \mathfrak{B}|(\bigcup_{i \leq j} N(b_i, 3^{r-j}))$$

with an isomorphism mapping a_i to b_i.

When $r = j$, $a_i \mapsto b_i$.
Using this strong induction hypothesis you get the result.

Methodology

$3'$ $\mathfrak{A}_r$ is d-equivalent to $\mathfrak{B}_r$ for some $d \geq 3^r$.

This is sound but not complete. To see that it is not complete, consider the cardinality of linear orders.

Note: In the definition of d-equivalence, you only need the same number of points up to a certain level.

We can do the applications by picture now.

Application 4

Connectivity is not FO on finite graphs.

Let $\mathfrak{A}_r$ be the circuit with $4d$ nodes. Let $\mathfrak{B}_r$ be two disjoint circuits each with $2d$ nodes. $N(a, d)$ is the path with $2d - 1$ nodes. So $\mathfrak{A}_r$ is d-equivalent to $\mathfrak{B}_r$.

Application 5

2 colorability, i.e., there are no odd cycles, is not FO on finite graphs.

Let $\mathfrak{A}_r$ be the circuit with $6d$ nodes. Let $\mathfrak{B}_r$ be two disjoint circuits each with $3d$ nodes. Let $d = 3^r$.

Application 6

Acyclicity is not FO on finite graphs.

Let $\mathfrak{A}_r$ be the path with $4d$ nodes. Let $\mathfrak{B}_r$ be a path with $2d$ nodes and a circuit with $2d$ nodes.

Exercise: show planarity and Hamiltonicity are not FO.

FO can only express properties which are local. Theorem3 says that if we can't tell structures apart at the local level, then we can't tell them apart FO.

Extending the Methodology to Second-Order Logic

SO∃ has formulas of the form $\exists S_{1\searrow m_1} \ldots \exists S_{k\searrow m_k}\ \varphi_{\searrow r}(m_1, \ldots, m_k, r)$
Fagin (1972) showed that NP=SO∃ on finite structures.

EF Games for SO∃

EF Game with parameters $(m_1, \ldots, m_k, r)$ on $\mathfrak{A}, \mathfrak{B}$.

Part I
Spoiler picks ${S_1}^{\mathfrak{A}}, \ldots, {S_k}^{\mathfrak{A}}$ relations of arities $m_1, \ldots, m_k$ on $\mathfrak{A}$.
Duplicator picks ${S_1}^{\mathfrak{B}}, \ldots, {S_k}^{\mathfrak{B}}$ on $\mathfrak{B}$.
(Spoiler stays on $\mathfrak{A}$ and Duplicator stays on $\mathfrak{B}$.)

Part II
They play r-move EF game on $(\mathfrak{A}, {S_1}^{\mathfrak{A}}, \ldots, {S_k}^{\mathfrak{A}})$ and $(\mathfrak{B}, {S_1}^{\mathfrak{B}}, \ldots, {S_k}^{\mathfrak{B}})$.

Theorem 4 *TFAE*

1. *Duplicator wins EF game with parameters* $(m_1, \ldots, m_k, r)$ *on* $\mathfrak{A}, \mathfrak{B}$.
2. *Every SO∃ sentence* $\exists S_{1\searrow m_1} \ldots \exists S_{k\searrow m_k}\ \varphi_{\searrow r}$ *which is true on* $\mathfrak{A}$ *is also true on* $\mathfrak{B}$.

Methodology for definability in SO∃

Corollary 3 *Given* $\mathcal{C}$*, a class of structures,* Q*, a Boolean query on* $\mathcal{C}$*, to show* Q *is not SO∃ definable on* $\mathcal{C}$ *it suffices to show that for every* $(m_1, \ldots, m_k, r)$ *there are* $\mathfrak{A}$,$\mathfrak{B}$ *such that*

1. $\mathfrak{A} \in \mathcal{C}$, $\mathfrak{B} \in \mathcal{C}$.
2. $Q(\mathfrak{A}) = 1$, $Q(\mathfrak{B}) = 0$.
3. *Duplicator wins EF game with parameters* $(m_1, \ldots, m_k, r)$ *on* $\mathfrak{A}, \mathfrak{B}$.

This methodology is sound and complete.

Methodology to show NP≠coNP

Corollary 4 *TFAE*

1. *NP≠coNP*
2. *For every* $(m_1, \ldots, m_k, r)$ *there are finite graphs* $\mathfrak{A}, \mathfrak{B}$ *such that*
   (a) $\mathfrak{A}$ *is not 3-colorable,* $\mathfrak{B}$ *is 3-colorable.*

(b) Duplicator wins EF game with parameters $(m_1, \ldots, m_k, r)$ *on* $\mathfrak{A}, \mathfrak{B}$.

Proof:

Previous Corollary and NP=SO$\exists$ and 3-colorability is NP-complete.

This brings out the combinatorial difficulties of NP$\neq$coNP.
Fagin worked on making it easier for Duplicator to win.

Ajtai-Fagin Games for a query Q with parameters $(m_1, \ldots, m_k, r)$

Given $\mathcal{C}$, a class of structures, Q, a Boolean query on $\mathcal{C}$,

Part I
Duplicator selects $\mathfrak{A} \in \mathcal{C}$ such that $Q(\mathfrak{A}) = 1$.
Spoiler selects relations ${S_1}^{\mathfrak{A}}, \ldots, {S_k}^{\mathfrak{A}}$ of arities $m_1, \ldots, m_k$ on $\mathfrak{A}$.
Duplicator selects $\mathfrak{B} \in \mathcal{C}$ such that $Q(\mathfrak{B}) = 0$ and relations ${S_1}^{\mathfrak{B}}, \ldots, {S_k}^{\mathfrak{B}}$ of arities $m_1, \ldots, m_k$ on $\mathfrak{B}$.

Part II
They play r-move EF game on $\mathfrak{A}, \mathfrak{B}$.

Theorem 5 *Given $\mathcal{C}$ and Q, a query on $\mathcal{C}$, TFAE*

1. *Q is not definable on $\mathcal{C}$ by any SO$\exists$ sentence with parameters* $(m_1, \ldots, m_k, r)$.
2. *Duplicator wins AF game for Q with parameters* $(m_1, \ldots, m_k, r)$.

The interesting direction is 1$\rightarrow$2. You argue by contradiction. Exercise.

Monadic SO

$\exists S_{1\searrow 1} \ldots \exists S_{k \searrow 1}\, \varphi_{\searrow r}$
We write $(1, \ldots, 1, r)$ as $\langle k, r\rangle$.

$\langle k, r\rangle$ is an AF game for Q with parameters $(\overbrace{1, \ldots, 1}^{k}, r)$.
We can prove lower bounds. Things are easier for the Duplicator.
$S_1, \ldots, S_k$ are sets. We can think of them as k colors on the nodes.
Make a color $C(a) = (1, 0, 0, 1, \ldots, 1)$

1 in the ith position if $a \in {S_i}^{\mathfrak{A}}$
0 in the ith position if $a \notin {S_i}^{\mathfrak{A}}$

This is a very local thing.

Application

Theorem 6 Fagin, 1975 *Connectivity is <u>not</u> monadic SO$\exists$ definable on finite graphs.*

We give a simpler proof.

Why do we care? Fact: Disconnectivity is monadic SO$\exists$ definable on finite graphs.

$$(\exists S)(S \neq \emptyset \wedge \overline{S} \neq \emptyset \wedge \forall x \forall y\, S(x) \wedge \overline{S}(y) \rightarrow \neg E(x, y)$$

Corollary 5 *monadic SO$\exists$ $\neq$ monadic SO$\forall$.*

Proof of theorem:
Use AF game for connectivity on finite graphs.
Let $\mathfrak{A}$ be a circuit. We have ${S_1}^{\mathfrak{A}}, \ldots, {S_k}^{\mathfrak{A}}$ which gives us a coloring of a point a, $C(a) = (1, 0, \ldots, 1)$. Let $\mathfrak{B}$ be two disjoint circuits.
In the expanded structures with colors assigned, $N(a, d)$ are paths with $2d - 1$ points together with colors $C(a_i)$ on each point in $N(a, d)$.

$$a-1 \quad \overset{C(a)}{a} \quad \overset{C(a+1)}{a+1} \quad a+2$$

There are 2^k different assignments of colors. There are $2d - 1$ points. So the number of d-types depends only on k and d. Fix k and d and take a large enough cycle.
No matter what ${S_1}^{\mathfrak{A}}, \ldots, {S_k}^{\mathfrak{A}}$ Spoiler plays, we can find $4d$ points of the same d-type.
Take such a cycle, then we have at least 2 points with the same d-type and distance of at least $2d$. Say the d-types of a_p and a_q are the same. Assume a_{p-1} is connected to a_p which is connected to a_{p+1}. Similarly, a_{q-1} is connected to a_q which is connected to a_{q+1}. In $\mathfrak{B}$, which consists of two disjoint cycles, the point corresponding to a_p is connected to the point corresponding to a_{q+1} and the point corresponding to a_q is connected to the point corresponding to a_{p+1}. The Duplicator copies the colors of the Spoiler.
The expanded structures have the same number of points of the same d-type for every d-type.

We have seen the following:

1. FO has limited expressive power on finite graphs.

2. SO pushes you to NP and beyond. SO$\exists$ =NP.

Question: What do you add to FO to enhance its expressive power but stay within P?
Hint: FO lacks a recursion mechanism on <u>finite</u> structures.
Answer: Add recursion to FO.

We can think of recursion as solutions to equations. For example:

$$\begin{aligned} f(0) &= 1 \\ f(n+1) &= (n+1)\, f(n) \end{aligned}$$

$$f = \lambda\, n\, .\, (n = 0 \rightarrow 1 \square\; n\, f(n-1))$$

Factorial is the "smallest" solution of this recursive specification.

Kleene - inductive definability
Use FO logic to specify queries recursively.

Transitive closure TC

$$S(a,b) \Leftrightarrow E(a,b) \vee \exists z(S(a,z) \wedge S(z,b)$$

TC is the "smallest" relation satisfying this specification.

FO formulas can be viewed as operators. $\varphi(x_1,\ldots,x_k,S_{\searrow k})$ over vocabulary $\sigma \cup \{S\}$. This induces operators on $\mathcal{P}(A^k)$ for every structure $\mathfrak{A}$ over σ.

$$\Phi : \mathcal{P}(A^k) \to \mathcal{P}(A^k)$$
$$\Phi(S) = \{(a_1,\ldots,a_k) \in A^k \mid \mathfrak{A} \models \varphi(x_1/a_1,\ldots,x_k/a_k,S)\}$$

Φ is <u>monotone</u> if $S_1 \subseteq S_2 \Rightarrow \Phi(S_1) \subseteq \Phi(S_2)$.

<u>Fact</u>: If $\varphi(x_1,\ldots,x_k,S)$ is positive in S then Φ is monotone. (Each occurrence of S is within an even number of $\neg$'s.)

Theorem 7 Knaster-Tarski *Every monotone operator has a least fixed point (smallest S such that $\Phi(S) = S$).*

Proof:
We do a bottom up evaluation of the least fixed point.

$$\Phi^1 = \Phi(\emptyset)$$
$$\Phi^{n+1} = \Phi(\Phi^n)$$

$$\Phi^1 \subseteq \Phi^2 \subseteq \ldots \Phi^n \subseteq \Phi^{n+1} \subseteq \ldots$$

inductively using monotonicity.

If we do this on a finite $\mathfrak{A}$, $\Phi^n \subseteq \Phi^{n+1} \subseteq A^k$. There is some $n_0 \leq \mid A^k \mid$ such that $\Phi^{n_0} = \Phi^{n_0+1} =$ the least fixed point.

This evaluation can be carried out in a polynomial number of steps.

Notation:
Given $\varphi(x_1,\ldots,x_k,S)$, k-ary positive, the least fixed point is $\varphi^\infty(x_1,\ldots,x_k)$.

Definition 2 *LFP logic = FO + least fixed point of positive FO formulas, closed under $\forall$, $\exists$, $\wedge$, $\vee$.*

We get that FO $\subsetneq$ LFP $\subseteq$ P.

For example, suppose $Q(x,y,S)$ is $E(e,y) \vee (\exists S)(S(x,z) \wedge S(z,y))$. Then $\varphi^\infty(x,y)$ is TC and $\forall x \forall y\, \varphi^\infty(x,y)$ is connectivity.

notes taken by Brenda J. Latka, latka@dimacs.rutgers.edu

Session III: Wednesday, August 16
Track A: Expressive Power of Logics

Phokion Kolaitis

Definition 1 FO+LFP *is the least set of formulae containing the formulae of first-order logic plus* $\varphi^\infty(\vec{x}, S)$ *for first-order* φ *positive in* S *and closed under* $\wedge$, $\vee$, $\forall$, *and* $\exists$.

Note that (so far) there is no nesting nor complement of lfps.

Some examples:

- Transitive Closure

$$\varphi(x, S) \equiv E(x, y) \vee (\exists z)[S(x, z) \wedge S(z, y)].$$

- Acyclicity

$$\psi(x, S) \equiv (\forall y)[E(y, x) \rightarrow S(y)].$$

  Following the bottom-up construction of the lfp:

$$\begin{aligned}
\Psi^1 = \Psi(\emptyset) &= \{x \mid x \text{ has in degree } 0\} \\
\Psi^2 &= \{x \mid (\forall y)[E(y, x) \rightarrow y \text{ has in degree } 0]\} \\
&\vdots \\
\Psi^\infty &= \{x \mid \text{ no path through } x \text{ leads to a cycle}\} \text{ (in finite graphs)}
\end{aligned}$$

  (In general, $\psi^\infty(x)$ picks out those x for which every path through x is well-founded.)

  With this definition of ψ^∞, then, $(\forall x)[\psi^\infty(x)]$ is acyclicity on finite graphs (which is complete for LOGSPACE).

- Path Systems ([2]—complete for P)

  Let $\mathfrak{A} = (V, A, R)$ where A is unary and R is ternary.

  We will interpret A as a set of axioms and R as an inference rule, where $R(x, y, z)$ iff R derives x from y and z. The set of theorems in this system can be defined inductively with

$$\psi(x, T) \equiv A(x) \vee (\exists y, z)[R(x, y, z) \wedge T(y) \wedge T(z)].$$

  Computing ψ^∞ is P-complete. Thus FO+LFP contains P-complete problems, although we will see that on all finite structures (rather than just those with order) it is not equal to P.

Some history

FO+LFP on finite structures is due originally to Chandra and Harel [1]. On infinite structures, though, it goes back to Kleene's *Hyperarithmetic Theory.*

Theorem 8 (See Chapter 2 in [4]) *On the structure* $\mathbb{N} = (N, +, \times)$ *(or even just* $(N, <)$*)* FO+LFP $=$ SO$\forall$.

Thus on $\mathbb{N}$ the analog of P (i.e., FO+LFP) is equivalent to the analog of NP (i.e., SO$\forall$).

Note that FO+LFP on $\mathbb{N}$ is not closed under complement; SO$\forall \neq$ SO$\exists$ on $\mathbb{N}$. This led Chandra and Harel [1] to ask whether FO+LFP on finite structures is closed under complement (and to suspect that it was not). Immerman [3] proved that it was. His proof is based on the *Stage Comparison Theorem.*

The Stage Comparison Theorem

Definition 2 *Let* $\varphi(\vec{x}, S)$ *be an* S*-positive first-order formula and* $\mathfrak{A}$ *a structure. Suppose* $\vec{a} \in \varphi^\infty(\vec{x})$*. Then* $|\vec{a}|_\varphi$ *denotes the smallest* m *such that* $\vec{a} \in \Phi^m$ *or* ∞ *if* $\vec{a} \notin \Phi^\infty$*.*

Definition 3 (Stage Comparison Queries) *Suppose* φ *and* ψ *are positive inductive formulae. Then:*

$$\vec{a} \leq^*_{\varphi,\psi} \vec{b} \overset{\text{def}}{\iff} a \in \varphi^\infty \text{ and } (\vec{b} \notin \psi^\infty \text{ or } |\vec{a}|_\varphi \leq |\vec{b}|_\psi).$$
$$\vec{a} <^*_{\varphi,\psi} \vec{b} \overset{\text{def}}{\iff} a \in \varphi^\infty \text{ and } (\vec{b} \notin \psi^\infty \text{ or } |\vec{a}|_\varphi < |\vec{b}|_\psi).$$

Examples

- Let φ be slow TC. Then $(a_1, b_1) <^*_{\varphi,\varphi} (a_2, b_2)$ iff there is a path from a_1 to b_1 and it is shorter than any path from a_2 to b_2. I.e., the stage comparison query for TC is the distance query.

- Let φ be the path system query of Cook. Then $s_1 <^*_{\varphi,\varphi} s_2$ iff s_1 is a theorem and is provable in less depth (i.e., depth of the proof tree) than any proof of s_2.

Theorem 9 (Moschovakis [4]) Stage Comparison Theorem. $\leq^*_{\varphi,\psi}$ *and* $<^*_{\varphi,\psi}$ *are definable in* FO+LFP*. In fact, they are lfp of positive first-order formulae.*

Proof (for $\leq^*_{\varphi,\varphi}$)**:** We are looking for a positive first-order formula $\chi(\vec{x}, \vec{y}, U)$ such that $\vec{x} \leq^*_{\varphi,\varphi} \vec{y} \Leftrightarrow \chi^\infty(\vec{x}, \vec{y})$.

The idea is to find a recursive specification satisfied by $\leq^*_{\varphi,\varphi}$ which can be shown to be sufficient to actually capture it.

Let $\Phi^{<m}$ denote $\bigcup_{l<m} \Phi^l$. Then

$$\begin{aligned}\vec{x} \leq^*_{\varphi,\varphi} \vec{y} &\Leftrightarrow \Phi^{|\vec{y}|_\varphi}(\vec{x})\\ &\Leftrightarrow \varphi(\vec{x}, \Phi^{<|\vec{y}|_\varphi})\\ &\Leftrightarrow \varphi(\vec{x}, \{\vec{x'} \mid \vec{x'} <^*_{\varphi,\varphi} \vec{y}\})\end{aligned}$$

Under the assumption that $\vec{x'} \in \varphi^\infty$ (which holds here by definition of $<^*_{\varphi,\varphi}$):

$$\begin{aligned}\vec{x'} <^*_{\varphi,\varphi} \vec{y} &\Leftrightarrow \neg\Phi^{|\vec{x'}|_\varphi}(\vec{y})\\ &\Leftrightarrow \neg\varphi(\vec{y}, \{\vec{y'} \mid \vec{y'} <^*_{\varphi,\varphi} \vec{x'}\})\\ &\Leftrightarrow \neg\varphi(\vec{y}, \{\vec{y'} \mid \neg(\vec{x'} \leq^*_{\varphi,\varphi} \vec{y'})\})\end{aligned}$$

Combining these

$$\vec{x} \leq^*_{\varphi,\varphi} \vec{y} \Leftrightarrow \varphi(\vec{x}, \{\vec{x'} \mid \neg\varphi(\vec{y}, \{\vec{y'} \mid \neg(\vec{x'} \leq^*_{\varphi,\varphi} \vec{y'})\})\}).$$

This gives us our recursive specification of U. Let

$$\chi(\vec{x}, \vec{y}, U) \equiv \varphi(\vec{x}, \{\vec{x'} \mid \neg\varphi(\vec{y}, \{\vec{y'} \mid \neg U(\vec{x'}, \vec{y'})\})\}).$$

Our claim is that $\leq^*_{\varphi,\varphi} = \chi^\infty$.

Because $\leq^*_{\varphi,\varphi}$ is a fixpoint of the recursive specification, $\chi^\infty \subseteq \leq^*_{\varphi,\varphi}$.

For the $\supseteq$ direction we can show, by induction on $|\vec{x}|_\varphi$, that if $\vec{x} \leq^*_{\varphi,\varphi} \vec{y}$ then $\chi^\infty(\vec{x}, \vec{y})$. □

Note that if φ is existential then χ is $\exists\forall$. This actually has to be the case since the stage comparison query of TC is the distance query and the distance query is not preserved under extensions while all existential queries are.

Theorem 10 (Immerman [3]) FO+LFP *is closed under complement on the class of* all *finite structures.*

Proof: It suffices to show that if $\varphi(\vec{x}, S) \in$ FO then $\neg\,\varphi^\infty \in$ FO+LFP.

Since the theorem is false on the class of all structures (by the Kleene-Spector theorem) the proof will necessarily exploit a property peculiar to finite structures. We will use the fact that, for finite structures, the closure ordinal of the lfp is a successor ordinal. Consequently, there is a non-empty last stage of its bottom-up construction.

Let Max_φ denote this last stage:

$$\mathrm{Max}_\varphi \stackrel{\mathrm{def}}{=} \{\vec{x} \in \varphi^\infty \mid (\forall\vec{y})[\vec{y} \in \varphi^\infty \rightarrow |\vec{y}|_\varphi \leq |\vec{x}|_\varphi]\}.$$

Given that Max_φ is non-empty, $\vec{x} \notin \varphi^\infty$ iff there exists some $\vec{y}$ such that $\mathrm{Max}_\varphi(\vec{y})$ and $\vec{y} <^*_{\varphi,\varphi} \vec{x}$. (The $\Rightarrow$ direction is true, of course, only in case Max_φ is non-empty.) Therefore, it suffices to show that Max_φ is in FO+LFP.

We might define this as

$$\vec{x} \in \mathrm{Max}_{\varphi} \Leftrightarrow (\forall \vec{y})[\neg\, \varphi^{\infty}(\vec{y}) \vee \vec{y} \leq^{*}_{\varphi,\varphi} \vec{x}]$$

but, as our goal is to define $\neg\, \varphi^{\infty}$, this introduces a circularity. Rather, we will use a *one-step delay* and the stage comparison query to find Max_{φ}.

Exercise: Given positive inductive $\varphi(\vec{x}, T)$, find $\psi(\vec{x}, \vec{y}, \vec{z}, T)$ where

1. $\varphi^{\infty}(\vec{x}) \Leftrightarrow \psi^{\infty}(\vec{x}, \vec{x}, \vec{x})$ and
2. if $\vec{x} \in \varphi^{\infty}$ and $|\vec{x}|_{\varphi} = m$ then $|\vec{x}, \vec{x}, \vec{x}|_{\psi} = m + 1$.

Claim 1

$$\vec{x} \in \mathrm{Max}_{\varphi} \Leftrightarrow \vec{x} \in \varphi^{\infty} \;\; \textit{and} \; (\forall \vec{z})[\vec{x} <^{*}_{\varphi,\varphi} \vec{z} \rightarrow (\vec{x}, \vec{x}, \vec{x}) <^{*}_{\psi,\varphi} \vec{z}].$$

For the $\Rightarrow$ direction, note that if $\vec{x}$ is in the last stage then both $\vec{x} \in \varphi^{\infty}$ and $(\vec{x}, \vec{x}, \vec{x}) \in \psi^{\infty}$, and $\vec{x} <^{*}_{\varphi,\varphi} \vec{z}$ implies that $\vec{z} \notin \varphi^{\infty}$. Consequently $(\vec{x}, \vec{x}, \vec{x}) <^{*}_{\psi,\varphi} \vec{z}$.

For the converse, assume for contradiction that the right-hand side holds but $\vec{x} \notin \mathrm{Max}_{\varphi}$. Then $\vec{x}$ is in φ^{∞} but not in the last stage. It follows that some $\vec{z}$ is added at the stage immediately following $\vec{x}$, i.e., there is some $\vec{z} \in \varphi^{\infty}$ such that $|\vec{z}|_{\varphi} = |\vec{x}|_{\varphi} + 1$.

But then $\vec{z}$ satisfies $\vec{x} <^{*}_{\varphi,\varphi} \vec{z}$ but fails to satisfy $(\vec{x}, \vec{x}, \vec{x}) <^{*}_{\psi,\varphi} \vec{z}$ contradicting our assumption.

Thus the claim holds, but the right-hand side includes a negative instance of the FO+LFP query $\vec{x} <^{*}_{\varphi,\varphi} \vec{z}$ (in the antecedent of the implication). Note though, since $\vec{x} \in \varphi^{\infty}$, it is the case that $\neg(\vec{x} <^{*}_{\varphi,\varphi} \vec{z}) \Leftrightarrow \vec{z} \leq^{*}_{\varphi,\varphi} \vec{x}$ and the negation can be eliminated. □

This construction gives a compound FO+LFP formula for $\neg\, \varphi^{\infty}$ that involves three FO+LFP formulae as subformulae. It is reasonable to ask whether these can be combined into the lfp of a single positive FO formula. In general the answer is no.

Proposition 1 $\neg\mathrm{TC}$ *is not* $\varphi^{\infty}(x, y)$ *for any positive* FO *formula* φ*.*

That is, closure with the Boolean connectives and first-order quantifiers is necessary.

Proof (Moschovakis)**:** Suppose, for contradiction, that $\neg\mathrm{TC}(x, y) \Leftrightarrow \varphi^{\infty}(x, y)$ for some positive FO φ.

A graph G is disconnected iff $(\exists x, y)[\neg\mathrm{TC}(x, y)]$ and, by assumption then, iff $(\exists x, y)[\varphi^{\infty}(x, y)]$, i.e. iff Φ^{∞} is non-empty. Note that Φ^{∞} is non-empty iff Φ^{1} is non-empty. Therefore G is disconnected iff $(\exists x, y)[\varphi(x, y, \emptyset)]$. This last is a FO formula, but connectivity is not FO definable—a contradiction. □

Normal Form Theorems for FO+LFP

FO+LFP admits normal forms of the following sorts:

Theorem 11 (Normal Forms for FO+LFP**)**

1. *Assume the vocabulary has two constants c and d, and that $c \neq d$ in all structures. Then every* FO+LFP *query can be expressed in the form*

$$Q(\vec{x}) \Leftrightarrow \varphi^{\infty}(c, \ldots, c, d, \ldots, d, \vec{x}).$$

2. *Without constants, every* FO+LFP *query can be expressed in the form*

$$Q(\vec{x}) \Leftrightarrow (\exists s, t)[s \neq t \wedge \varphi^{\infty}(s, \ldots, s, t, \ldots, t, \vec{x})].$$

Thus, FO+LFP is quite robust on finite structures—in particular, it is closed under

- negations (as shown above) and
- iterations of lfp (see Session IV).

Since connectivity is not FO even for ordered finite graphs these normal-forms are optimal even for ordered finite structures.

References

[1] A. Chandra and D. Harel. Computable queries for relational databases. *Journal of Computer and System Sciences*, 21:156–178, 1980.

[2] S. Cook. Path systems and language recognition. *2nd ACM Symposium on Theory of Computing*, 1970, pp. 70–72.

[3] N. Immerman. Relational Queries Computable in Polynomial Time. *Information and Control* 68 (1986), 86-104. A preliminary version appeared in *14th ACM STOC Symp.* (1982), 147-152.

[4] Y. N. Moschovakis. *Elementary Induction on Abstract Structures.* North Holland, 1974.

notes taken by Jim Rogers, jrogers@linc.cis.upenn.edu

Session IV: Thursday, 17 August 1995
Track A: Expressive Power of Logics
Phokion Kolaitis

1 Review

We have been talking about FO+LFP on the class of all finite structures, defined in the most economical way possible. That is:

- FO+LFP = FO logic with least fixed points of S-positive FO formulas
$$\phi(x_1, \ldots, x_k, S),$$
closed under $\wedge, \vee, \forall, \exists$.

We saw yesterday that this is closed under complementation on the class of all finite structures (*cf.* the Immerman Complementation Theorem).

2 Today's Plan

- More closure properties of FO+LFP.
- Properties of inflationary fixed point logic: FO+IFP.
- Examining the expressive power of FO+LFP, by embedding FO+LFP in the infinitary logic, $L^{\omega}_{\infty,\omega}$.

3 Iterative Systems of positive FO formulas

(Also called *mutual recursion for* FO+LFP)
Given the system:

$$\left.\begin{array}{c} \phi_1(\vec{x}_1, S_1, \ldots, S_l) \\ \vdots \\ \phi_l(\vec{x}_l, S_1, \ldots, S_l) \end{array}\right\} \begin{array}{l} \phi_i \text{ is positive in } S_1, \ldots, S_l; \\ |x_i| = \text{the arity of } S_i, i = 1, \ldots, l \end{array}$$

Iterate it. That is, define the stages of the system by simultaneous induction:

$$\begin{array}{rcl} \Phi_1^0(\vec{x}_1) & = & \phi_1(\vec{x}_1, \emptyset, \ldots, \emptyset) \\ & \vdots & \\ \Phi_l^0(\vec{x}_l) & = & \phi_l(\vec{x}_l, \emptyset, \ldots, \emptyset) \\ & & \\ \Phi_1^{m+1}(\vec{x}_1) & = & \phi_1(\vec{x}_1, \Phi_1^m, \ldots, \Phi_l^m) \\ & \vdots & \\ \Phi_l^{m+1}(\vec{x}_l) & = & \phi_l(\vec{x}_l, \Phi_1^m, \ldots, \Phi_l^m) \end{array}$$

For finite $\mathfrak{A}$, there is some m_0 such that for all $i = 1, \ldots, l$:

$$\Phi_i^{m_0} \equiv \Phi_i^{m_0+1}$$

The **least fixed point of the system** $(\phi_1, \ldots, \phi_l)$ is $(\Phi_1^{m_0}, \ldots, \Phi_l^{m_0})$.
Example: Let

$$\begin{aligned} \phi_1(x, y, S_1, S_2) &\equiv E(x, y) \vee (\exists z)(E(x, z) \wedge S_1(z, y)) \\ \phi_2(x, S_1, S_2) &\equiv S_1(x, x) \end{aligned}$$

Then,

$$\begin{aligned} \Phi_1^\infty &\equiv \text{the transitive closure of } E \\ \Phi_2^\infty &\equiv \{x \mid x \text{ is on a cycle}\} \end{aligned}$$

4 The Normal Form Theorem

Theorem 12 *The following are equivalent:*

1. *Q is a* FO+LFP *definable query.*

2. *There is a system $(\phi_1, \ldots, \phi_l)$ of positive* FO *formulas such that $Q \equiv \Phi_l^\infty$.*

Remarks:

1. This result assumes the language has equality.

2. Instead of closure by $\wedge, \vee, \exists, \forall$, you can get a system.

3. $l \geq 2$ is necessary. (In fact, $l = 2$ suffices.)

Example: Let $G = (V, E)$ be a finite query and $s, t \in V$. Consider the query:

$$Q(s, t) \Leftrightarrow \text{there exists a simple path of even length from } s \text{ to } t.$$

- **Fortune, Hopcroft, Wiley (1982)** showed, on directed graphs, this query is NP-complete. This can be seen by reducing Q to the NP-complete "2 disjoint paths" query by an essentially FO reduction (*i. e.* given two disjoint paths from s_1 to s_2 and s_3 to s_4, double all the edges, add the edge (s_2, s_3) and a new node t and edge (s_4, t)).

- **Yannakakis**: On undirected graphs, this query is in FO+LFP (hence, in P). He gave the system:

$$\begin{aligned} \phi_1(x, y, z, S_1, S_2, S_3) &\equiv (E(x, y) \wedge z \neq x \wedge z \neq y) \\ &\quad \vee (\exists w)(E(x, w) \wedge S_1(w, y, z) \wedge z \neq x) \\ \phi_2(x, y, z, S_1, S_2, S_3) &\equiv E(x, y) \\ &\quad \vee (\exists w)(S_3(x, w) \wedge E(w, y) \wedge S_1(x, w, y)) \\ \phi_3(x, y, z, S_1, S_2, S_3) &\equiv (\exists w)(S_2(x, w) \wedge E(w, y) \wedge S_1(x, w, y)) \end{aligned}$$

  Exercise: Prove, over the class of undirected graphs, that:

– $\Phi_1^\infty(x, y, z)$ defines "is there a simple path from x to y, avoiding z"

– $\Phi_2^\infty(x, y)$ defines "is there a simple path of odd length from x to y"

– $\Phi_3^\infty(x, y)$ defines "is there a simple path of even length from x to y"

5 Inflationary Fixed Points

Question: How do we go beyond FO+LFP, but still stay in P?
Answer: Add seemingly more powerful iteration principles.

- **Aczel/Richter** introduced non-monotone inductive definabilty.

- **Gurevich** explored the idea further, under the name of **inflationary fixpoints**, on finite structures.

Idea: Carry along what you know already from stage to stage.
Given an operator $\Phi : \mathcal{P}(A^k) \to \mathcal{P}(A^k)$. Define the **inflationary stages** as follows:

$$\begin{aligned} \Phi^1 &= \Phi(\emptyset) \\ \Phi^{m+1} &= \Phi(\Phi^m) \cup \Phi^m \end{aligned}$$

By construction, $\Phi^1 \subseteq \Phi^2 \subseteq \cdots \Phi^m \subseteq \Phi^{m+1} \subseteq \cdots A^k$. On finite structures, there is a m_0 such that

$$\Phi^{m_0} = \Phi^{m_0+1}.$$

Call Φ^{m_0} the **inflationary fixpoint of Φ.**
Aczel/Richter showed that on $\mathbb{N} = (N, \leq)$,

$$\mathsf{FO} + \mathsf{LFP} \subset \mathsf{FO} + \mathsf{IFP}.$$

On finite structures,

$$\mathsf{FO} \subset \mathsf{FO} + \mathsf{LFP} \subseteq \mathsf{FO} + \mathsf{IFP} \subseteq \mathsf{P}.$$

Theorem 13 (Stage Comparison Theorem for FO+IFP): *The stage comparison relations, $\leq^*_{\phi,\psi}$ and $<^*_{\phi,\psi}$ are in* FO+IFP.

Proof: Let

$$\begin{aligned} \phi_1(\vec{x}, S_1, S_2, S_3) &\equiv \phi(\vec{x}, S_1) \\ \phi_2(\vec{y}, S_1, S_2, S_3) &\equiv \psi(\vec{y}, S_2) \\ \phi_3(\vec{x}, \vec{y}, S_1, S_2, S_3) &\equiv S_1(\vec{x}) \wedge \neg S_2(\vec{y}) \end{aligned}$$

The FO+IFP solution to the system gives $\Phi_3^\infty =<^*_{\phi,\psi}$.
$\leq^*_{\phi,\psi}$ follows similarly. $\dashv$

Theorem 14 Gurevich-Shelah (Ann. Pure and Appl. Logic ,1986) *On finite structures,*

$$\mathsf{FO} + \mathsf{LFP} = \mathsf{FO} + \mathsf{IFP}.$$

Proof: Uses the stage comparison relations and the Complementation Theorem of Immerman. ⊣

Leivant (Inf. & Control,1988) also shows this in a very short proof. His single system includes the definitions of the stage comparison relations.

Abiteboul, Vianu (1988) show that on finite structures,

$$\mathsf{FO} + \mathsf{LFP} = \mathsf{FO}{+}\exists\mathsf{IFP}.$$

We will illustrate the proof by giving a $\mathsf{FO}{+}\exists\mathsf{IFP}$ formula for $\neg\mathsf{TC}$. Let

$$\begin{array}{rcl}
\phi_1(x, y, S_1, S_2, S_3, S_4) & \equiv & E(x, y) \vee (\exists z)(E(x, z) \wedge S_1(z, y)) \\
\phi_2(x, y, S_1, S_2, S_3, S_4) & \equiv & S_1(x, y) \\
\phi_3(x, y, S_1, S_2, S_3, S_4) & \equiv & S_2(x, y) \wedge \\
 & & (\exists x', y', z')(\neg S_1(x', y') \wedge E(x', z') \wedge S_1(z', y')) \\
\phi_4(x, y, S_1, S_2, S_3, S_4) & \equiv & \neg S_2(x, y) \wedge (\exists x', y')(S_2(x', y') \wedge \neg S_3(x', y'))
\end{array}$$

Note that:

1. Φ_2^m calculates the "one-step delay" of Φ_1^m (*i. e.* $\Phi_2^{m+1} = \Phi_1^m$).
2. Φ_4^m stays empty as long as $\Phi_2^m = \Phi_3^m$. When $\Phi_2^m \neq \Phi_3^m$, Φ_4^{m+1} is the complement of the transitive closure.

This yields:

- $\Phi_1^\infty = \mathsf{TC}$.
- $\Phi_2^\infty = \mathsf{TC}$ with a one step delay.
- $\Phi_3^\infty = \mathsf{TC} - \{(a, b) \in \mathsf{TC} \mid \text{the distance from } a \text{ to } b \text{ is maximal}\}$.
- $\Phi_4^\infty = \neg\mathsf{TC}$.

So, we have:

$$\mathsf{FO}{+}\exists\mathsf{LFP} \subset \mathsf{FO} + \mathsf{LFP} = \mathsf{FO}{+}\exists\mathsf{IFP} = \mathsf{FO} + \mathsf{IFP}$$

$\mathsf{FO}{+}\exists\mathsf{LFP} \subset \mathsf{FO} + \mathsf{LFP}$ follows from the expressibility of "acyclicity" in the latter but not the former.

Question: how can we show $FO + LFP \subset P$? Even cardinality is not in $FO + LFP$.

6 Infinitary Logic

Let us give a new definition of the Transitive Closure.

$$\begin{aligned} TC(x,y) &\equiv \text{ there is a path from } x \text{ to } y \\ P_m(x,y) &\equiv \text{ there is a path of length} \leq m \text{ from } x \text{ to } y \\ TC(x,y) &\equiv \textstyle\bigvee_{m\geq 1} P_m(x,y) \end{aligned}$$

This last definition of the transitive closure brings us to infinitary logic. Henkin and Tarski introduced infinitary logic. There are two ways to extend logic into infinitary logic. One is to have infinite disjunctions and conjunctions, and the other is to have infinite strings of quantifiers.

The notation $L_{\infty,\omega}$ means that we have infinite $\vee/\wedge$'s (indicated by the symbol ∞), but the quantifier strings are finite (indicated by the symbol ω).

So if the ϕ_i's are in $L_{\infty,\omega}$, then also $\wedge\phi_i$ and $\vee\phi_i$ are.

But $L_{\infty,\omega}$ is too strong: on the class of all finite structures $L_{\infty,\omega}$ defines all queries. This is how: take a query. Look at all the finite structures that satisfy it under isomorphism. Express each in FO and write the disjunction of all the formulas expressing the models of the queries.

But to do what we just described, we need many variables. To restrict the power of $L_{\infty,\omega}$, we are going to bound the number of variables that we can use. $L^k_{\infty,\omega}$ means that we are allowed to use at most k different variables, even though each variable can be used infinitely many times. Finally, $L^\omega_{\infty,\omega} = \bigcup L^k_{\infty,\omega}$ is infinitary logic with finitely many variables.

Example: Transitive closure is in $L^\omega_{\infty,\omega}$

$$\begin{aligned} P^m(x,y) &\equiv \text{ there is a path of length} \leq m \text{ from } x \text{ to } y \\ &\equiv \exists z_1 \ldots \exists z_{m-1}(E(x,z_1) \wedge E(z_1,z_2) \wedge \cdots \wedge E(z_{m-1},y)) \end{aligned}$$

But we can do it using 3 variables, by reusing variables.

$\phi_m(x,y)$ can be written in $L^3_{\omega,\omega}$.

$$\begin{aligned} \psi_1(x,y) &\equiv E(x,y) \\ \psi_{m+1}(x,y) &\equiv \exists z(E(x,z) \wedge \exists x(z = x \wedge \psi_m(x,y)) \end{aligned}$$

So the Transitive Closure is $L^3_{\infty,\omega}$ definable.

Theorem 15 *On all finite structures,* $FO + LFP \subset L^\omega_{\infty,\omega}$.

The former theorem can be proven by noting that $L^\omega_{\infty,\omega}$ can express non recursive queries. Also by noting that $FO + LFP$ cannot express even cardinality.

So, what is the advantage of $L^\omega_{\infty,\omega}$ over $FO + LFP$? The advantage is definability of $L^k_{\infty,\omega}$. $L^k_{\infty,\omega}$ can be easily characterized by pebble games.

7 k-pebble Games

Say that the spoiler has pebbles: $r_1, \cdots, r_k$
and the duplicator has pebbles: $g_1, \cdots, g_k$
Also there are structures A and B.

The spoiler places or moves pebble r_i on a element of A or B. The duplicator places or moves pebble g_i on an element of B or A (the other structure).

The spoiler wins if at some point the mapping $a_i \to b_i$ is not a partial isomorphism, where a_i and b_i are pebbled by $[r_i, g_i]$.

Duplicator wins if he can play so that at any point in time the partial isomorphism is preserved.

Example: The two structures are k disconected nodes, and $k+1$ disconnected nodes.

$$\bullet \bullet \cdots \bullet \bullet$$

$$\bullet \bullet \cdots \bullet \bullet \bullet$$

The spoiler wins the $k+1$-pebble game and the duplicator wins the k-pebble game. The same as with the EF game.

Example: The first structure is a graph consisting of a chain of n elements. The second is a chain of m elements.

$$\bullet \to \bullet \to \bullet \cdots \bullet \to \bullet$$

$$\bullet \to \bullet \to \bullet \cdots \bullet \to \bullet$$

If $m \neq n$ then the spoiler wins the 2-pebble game. The spoiler does that by forcing a walk over the graph.

Example: One structure is a loop of some length, and the other is two loops of half the length of the first loop. The spoiler in this case wins the 4-pebble game.

This is how they will play: the spoiler puts pebble s_1 in one of the loops of the second structure. Then, the duplicator puts pebble d_1 in a node of the first structure. The spoiler puts pebble s_2 in a different loop than he played before of the second structure. Then the duplicator places d_2 in the first structure as far away from d_1 as possible. From that point on the spoiler starts placing pebbles in the first structure, say s_3 is places between d_1 and d_2. The duplicator will place pebble d_3 in one of the two loops of the second structure. Say that he places d_3 in the same loop with s_1. The the spoiler places s_4 between d_2 and s_3. At this point the pebble d_1 has been freed. The duplicator will loose because the distance from d_3 to one of s_1 or s_2 is infinite, but instead the distance from s_3 to both d_1 and d_2 is finite.

Theorem 16 *(Barwise-Immerman) The following are equivalent:*

1) Structures A and B satisfy the same sentences of $L^k_{\infty,\omega}$.

2) The duplicator wins the k-pebble game on A and B.

Corollary 6 *(Methodology) To show that a query Q is not $L^{\omega}_{\infty,\omega}$-defined on the class C of structures, it suffices to show that $\forall k \geq 1$ there are A_k and B_k such that*

1)$A_k \in C$ and $B_k \in C$

2)$Q(A_k) = 1$ and $Q(B_k) = 0$

3) The duplicator wins the k-pebble game on A_k and B_k.

Corollary 7 *Even cardinality is not in $L^{\omega}_{\infty,\omega}$, hence not in $FO + LFP$. This is by the first example in this section.*

Notes taken by M. L. Bonet (bonet@dimacs.rutgers.edu) and K. St. John (stjohn@math.upenn.edu).

Session 5: Friday, August 18
Track A: Expressive Power of Logics

Phokion Kolaitis

Today:

1. A closer look at the $L^{\omega}_{\infty,\omega}$ k-pebble game.
2. Partial Fixpoint Logic.
3. The Abiteboul-Vianu Theorem, regarding Partial Fixpoint Logic.
4. Reconciliation of Track A and B of the tutorial.
5. Is there a logic for `PTIME`?

Reminder:
$L^{\omega}_{\infty,\omega}$: infinite $\bigwedge$ and $\bigvee$, and at most k distinct variables.

Theorem 17 *TFAE:*

1. $\mathcal{U} \equiv^{k}_{\infty,\omega} \mathcal{L}$ ($\mathcal{U}$ and $\mathcal{L}$ satisfy the same $L^{\omega}_{\infty,\omega}$ sentences.)

2. The duplicator has a winning strategy for the k-pebble game on $\mathcal{U}$ and $\mathcal{L}$.

Moreover, if $\mathcal{U}$ and $\mathcal{L}$ are finite:

3. $\mathcal{U} \equiv^{k}_{\omega,\omega} \mathcal{L}$ ($\mathcal{U}$ and $\mathcal{L}$ satisfy the same FO sentences.)

Methodology: For $\mathcal{C}$ a class of structures and Q a boolean query on $\mathcal{C}$, to show Q is not $L^{\omega}_{\infty,\omega}$-definable on $\mathcal{C}$ it suffices to show that for every $k \geq 1$ there are $\mathcal{U}_k$ and $\mathcal{L}_k$ such that:

1. $\mathcal{U}_k, \mathcal{L}_k \in \mathcal{C}$.
2. $Q(\mathcal{U}_k) = 1$ and $Q(\mathcal{L}_k) = 0$.
3. $\forall k$ the Duplicator has a winning strategy for the k-pebble game on $\mathcal{U}_k, \mathcal{L}_k$.

The methodology is sound and complete for $L^{\omega}_{\infty,\omega}$. It is sound but not complete for (FO + LFP). On finite structures (FO + LFP) $\subsetneq L^{\omega}_{\infty,\omega}$:

Why is (FO + LFP) $\subsetneq L^{\omega}_{\infty,\omega}$?

The containment follows from the fact that $\phi^{\infty} \equiv \bigvee_i \phi^i$, which we can express in $L^{\omega}_{\infty,\omega}$. That the containment is strict can be seen as follows:

Let L_n be the linear order on n elements, $n \geq 1$. Let ψ_m be the statement "there are at least m elements". On L_n ψ_m is definable in $L^{2}_{\omega,\omega}$:

$$\psi_m \equiv (\exists x)(\exists y)(x < y \wedge (\exists x)(y < x \wedge (\exists y)(x < y \wedge \ldots)) \ldots)$$

$\psi_m \wedge \neg\psi_m \equiv \chi_m \equiv$ "there are exactly m elements". Thus, on linear orders we can express arbitrary cardinality in $L^2_{\infty,\omega}$. Let $S \subseteq \mathcal{N}$.

$$L_n \models \bigvee_{m \in S} X_m \Leftrightarrow n \in S$$

$\bigvee_{m\in S} X_m \in L^2_{\infty,\omega}$, and thus we can express arbitrary predicates about the size of our structure in $L^2_{\infty,\omega}$. But we clearly can not do so in (FO + LFP) which we know is contained in `Ptime`.

Application: (of the k-pebble game) `Hamiltonian Cycle` is not $L^\omega_{\infty,\omega}$-definable on finite graphs.

Proof (de Rougement, 1983) Consider the undirected graph $\mathcal{U}_{n,m} = \overline{K}_n \times C_m$, where K_n is the complete undirected graph and C_m is the undirected m-cycle. In other words, there are n disconnected points that have all possible undirected edges to an m-cycle. $\mathcal{U}_{n,m}$ has a hamiltonian cycle if and only if $m \geq n$. Thus $\mathcal{U}_{k+1,k}$ is not hamiltonian while $\mathcal{U}_{k+1,k+1}$ is. But the Duplicator has a winning strategy for the k-pebble game on $\mathcal{U}_{k+1,k}$, $\mathcal{U}_{k+1,k+1}$.

Corollary 8 *Hamiltonicity is not (FO + LFP)-definable on finite graphs.*

- 3-colorability is not in $L^\omega_{\infty,\omega}$ [Dawar].
- Question: Is Planarity in $L^\omega_{\infty,\omega}$?

Side note: Fact (from Track B (Immerman) Lecture 4) On ordered structures $L^2_{\infty,\omega} = L^\omega_{\infty,\omega}$.

Fact 1 *For Q a boolean query on the class of all finite structures, TFAE:*

1. *Q is $L^k_{\infty,\omega}$-definable.*
2. *Q is the union of $\equiv^k_{\infty,\omega}$ equivalence classes, i.e., $Q = \bigcup_{\mathcal{L}\in\mathcal{D}}\{\mathcal{U} \mid \mathcal{L} \equiv^k_{\infty,\omega} \mathcal{U}\}$.*

In effect, Fact 1 says that every (FO + LFP) query is the union of $L^k_{\infty,\omega}$ equivalence classes for some k.

Theorem 18 *(Dawar, Lindell, and Weinstein, I&C 1995) On finite structures, each $\equiv^k_{\infty,\omega}$ equivalence class is definable by a sentence of $L^k_{\omega,\omega}$ (i.e., FO with k variables).*

For the proof they analyzed the k-pebble game, performing basically an analog of Scott analysis for finite structures.

Putting this together, we get a normal form theorem for $L^k_{\infty,\omega}$ on finite structures:

Theorem 19 (Normal form theorem) *Every $L^k_{\infty,\omega}$ sentence is equivalent on finite structures to a $L^k_{\infty,\omega}$ sentence of the form: $\bigvee_{m\geq 1}\psi_m$, $\psi_m \in L^k_{\omega,\omega}$, $\forall m \geq 1$.*

So far we have learned the following relationships for fixed point logics:

$$\text{FO} \subsetneq (\text{FO} + \text{LFP}) = (\text{FO} + \text{IFP}) \subsetneq \text{PTIME}$$

$$(\text{FO} + \text{LFP}) \subsetneq L^{\omega}_{\infty,\omega}$$

We can consider seemingly more powerful iteration mechanisms:
<u>Partial Fixpoint Logic</u>

Idea: "Iterate arbitrary FO formulas. If you hit a fixpoint, return it as the answer, else return a default value ($\emptyset$)." In other words, attempt to turn FO into a programming language by adding while looping.

History: Chandra and Harel defined (FO + While). Vardi (1982) showed that on ordered finite structures (FO + While) = PSPACE. Abiteboul-Vianu (1989) defined the much cleaner Partial Fixpoint Logic: Let $\phi(x_1, \ldots, x_k, S)$ be a first order formula with S k-ary. Given $\mathcal{U} = \langle A, \ldots \rangle$, ϕ induces a mapping $\Phi : P(A^k) \mapsto P(A^k)$ such that

$$\Phi(S) = \{(a_1, \ldots, a_k) \mid \mathcal{U} \models \phi(\frac{x_1, \ldots, x_k}{a_1, \ldots, a_k}, S)\}$$

We iterate Φ: $\Phi^1 = \Phi(\emptyset)$, $\Phi^{m+1} = \Phi(\Phi^m)$. There is no guarantee that the stages Φ^m increase, but $\Phi^m \subseteq A^k$ for all m, thus, for finite A, there is some m_0 such that either

1. $\Phi^{m_0} = \Phi^{m_0+1}$.

2. There is some $l > 1$ such that $\Phi^{m_0} = \Phi^{m_0+l}$, but for no smaller l does this hold.

We define the Partial Fixpoint of ψ:

$$\text{PFP}(\psi) = \psi^{\infty} = \begin{cases} \Phi^{m_0} & \text{if } \Phi^{m_0} = \Phi^{m_0+1} \\ \emptyset & \text{otherwise} \end{cases}$$

We let (FO + PFP) denote the logic with partial fixed points of arbitrary FO formulas, closed under $\wedge, \vee, \exists, \forall, \neg$.

Fact 2 *1. FO $\subsetneq$ (FO + LFP) $\subseteq$ (FO + PFP) $\subsetneq$ PSPACE*

2. (FO + PFP) $\subsetneq L^{\omega}_{\infty,\omega}$ (on finite structures).

Proof (1) (FO + PFP) $\subseteq$ PSPACE. Compute $\Phi^1, \Phi^2, \ldots \subseteq A^k$. If Φ^i reaches a fixpoint, it does so after at most $2^{|A|^k}$ iterations. After each iteration check if $\Phi^{m+1} = \Phi^m$. Maintain a counter (in PSPACE) that counts up to 2^{n^k}, where $n = |A|$. The containment is strict because evenness again can not be expressed in (FO + PFP), which follows from the containment in the second part of the fact, which in turn has the same proof as that of (FO + LFP) $\subsetneq L^{\omega}_{\infty,\omega}$.
■

Abiteboul-Vianu (1989): (FO + PFP) = (FO + While) on all finite structures.

<u>Corollary</u> (using Vardi (1982)) On ordered finite structures (FO + PFP) = PSPACE.

Example:

$$\phi(x,y,S) = (E(x,y) \wedge (\forall z)(\forall w)(\neg S(z,w))) \vee (\exists z)(E(x,z) \wedge S(z,y))$$

$\Phi^m \equiv$ "There is a path of length exactly m from x to y."

Exercise: Use this as a building block to show that the query "there is a (not necessarily simple) path from x to y whose length is a perfect square" is expressible in $(FO + PFP)$ (note that this can be expressed in simpler logics, but the expression here is particularly simple.)

Chandra and Harel's Problem (1982): Show that (FO + LFP) $\subsetneqq$ (FO + PFP) on all finite structures.

We now come to a result that provides some reconciliation between Tracks A and B of this tutorial. Remember that we began with Fagin's Theorem which showed that on finite structures Σ_1^1 = NP. Then in track A we continued to study the expressive power of logics on *all* finite structures, while in track B we studied descriptive complexity, i.e., the expressive power of logics on *ordered* structures, because of its tight connections to computational complexity. The follow important theorem of Abiteboul and Vianu to some extent brings these two tracks together.

Theorem 20 (Abiteboul & Vianu (STOC'91, I&C '95)) *TFAE:*

1. *P = PSPACE*
2. *(FO + LFP) = (FO + PFP) on* all *finite structures.*

Proof (See also Dawar, Lindell, and Weinstein (I&C'95) for a nice presentation of this result.)

2 $\Rightarrow$ 1: Obvious because (FO + LFP) = (FO + PFP) on all finite structures implies (FO + LFP) = (FO +PFP) on all ordered finite structures, thus (by Immerman and Vardi's results) P = PSPACE.

1 $\Rightarrow$ 2: If P = PSPACE then (FO + LFP)= (FO + PFP) (on all finite structures). The proof uses $L^k_{\infty,\omega}$ and the k-pebble game.

Key Lemma In (FO + LFP) we can define a total order on the k-types, where k-types are defined as follows: For $\mathcal{U}$ a finite structure, $(a_1,\ldots,a_k) \in A^k$, the k-type of $(a_1,\ldots,a_k)$ is defined as

$$\{\phi(x_1,\ldots,x_k) \mid \phi \in L^k_{\infty,\omega} \ \& \ \mathcal{U} \models \phi(\frac{x_1,\ldots,x_k}{a_1,\ldots,a_k})\}$$

k-types induce an equivalence relation on A^k, where the equivalence class of $(a_1,\ldots,a_k)$ is its equivalence class.

There are two key ideas in the proof of the Key Lemma:

1. "The spoiler wins the k-pebble game" can be expressed in (FO + LFP).
2. We can use a color refinement algorithm (cf., also Immerman-Lander'90) to order the k-types in (FO + LFP).

Suppose $\mathcal{C}$ is a class of finite structures. By taking quotients mod the $\equiv^k_{\infty,\omega}$ equivalence relation, we will map $\mathcal{C}$ to a set of quotient structures $\mathcal{C}^*_k$ in a way that induces a corresponding mapping on formulas such that $\phi \in (FO + LFP) \mapsto \phi^* \in (FO + LFP)$ and $\psi \in (FO + PFP) \mapsto \psi^* \in (FO + PFP)$.

We want to show that if Q is a (FO + PFP) query on a class $\mathcal{C}$ of finite structures, then (assuming P=PSPACE) Q is a (FO + LFP) query on $\mathcal{C}$. Q has a $\phi \in$ (FO + PFP) definition that uses at most k variables. We use the mapping above to go from Q to Q^*, and ϕ to ϕ^*. Now since P=PSPACE and since we can define an ordering on k-types in (FO + LFP), we have that $Q^* \in P$ and hence is definable in (FO + LFP) on $\mathcal{C}^*_k$ with some ψ. We thus get $Q \in (FO + LFP)$ on $\mathcal{C}$ by "pulling back" from ψ on the quotient structures to ψ' on the original structures. (There are several details to be filled in here, such as how to work with relations in the structure that have arity greater than k, etc., but these are the basic ideas). ■

A Logic for Ptime

We finally come to an open problem on which there has been a great deal of work:

Problem: "Is there a logic for PTIME?"

We need to make precise what this question means. Consider the following two queries on finite graphs:

1. Is the graph connected?

2. Is there a path from the "first" node to the "last"?

The first question is order independent: regardless of how the vertices of the graph are ordered the answer is the same. The second, on the other hand, does not even make sense unless we have an ordering on the vertices to tell what the "first" and "last" vertices are (alternatively, we could be presented the two vertices as constants in the structure. That would be order independent but that would really change the problem).

Let **Q** be the set of all order-independent queries on finite graphs.

Question: Is there a "logic" that captures exactly the order independent polynomial time queries, i.e., **Q**$\cap$ PTIME. Most generally stated (Chandra-Harel'1982) is the class of poly-time order-independent queries recursively enumerable[1]?

It is not hard to see that if Graph Canonization is in PTIME, then there is a logic for PTIME.

There is now a tutorial on a logic for PTIME which can be obtained by anonymous ftp from ftp.cse.ucsd.edu. Go to the pub/kolaitis directory. The file is icdt.tutorial.

Scribe: Kousha Etessami, etessami@dimacs.rutgers.edu

[1]This has to be via a listing from which we can retrieve the poly-time algorithm for each query, so say, can we give an r.e. listing of the clocked Turing machines that run in PTIME and accept exactly the PTIME order independent properties?