
Computability and Definability, Part 4

J.A. Makowsky

Department of Computer Science
Technion - Israel Institute of Technology
Haifa, Israel

janos@cs.technion.ac.il
<http://cs.technion.ac.il/~janos>

Course given the Technion, Haifa, Israel
Winter Semester, 2021/2

<http://www.cs.technion.ac.il/~janos/COURSES/236331-21>

- Location: Taub 8
- Time: Thursday 9:30-12:30 (including Tirgul)
- Office: Taub 628
- Office hours: Thursday 14:00 or by appointment

ZOOM

- makowsky@technion.ac.il is inviting you to a scheduled Zoom meeting.
- Topic: CS 236331
- Time: This is a recurring meeting: Thursday 9:30-12:30
- Join Zoom Meeting at:

<https://technion.zoom.us/j/91510301938>

Course requirements

- Prerequisites are NOT enforced.
Recommended prerequisites: Some familiarity with the concepts of Logic and sets, Automata and formal languages or Computability
- Homework
- Final project

Overview

Part 4:

- Translation schemes

Session 7 (December 16, 2021)

Part 4.1: Translation Schemes:

Main definitions and examples

- The framework of translation schemes
 - The induced maps
 - The fundamental lemma
 - Reductions
- The Museum of examples

Definition 1 (Translation Schemes Φ)

- Let τ and $\sigma = \{R_1, \dots, R_m\}$ be two vocabularies with $\rho(R_i)$ be the arity of R_i .
- Let $\mathcal{L}$ be a fragment of *SOL*, such as *FOL*, *MSOL*, $\exists$ *MSOL*, etc.
- Let $\Phi = \langle \phi, \psi_1, \dots, \psi_m \rangle$ be formulae of $\mathcal{L}(\tau)$ such that ϕ has exactly k distinct free first order variables and each ψ_i has $k\rho(R_i)$ distinct free first order variables.
We say that Φ is **k -feasible (for σ over τ)**.
- A k -feasible $\Phi = \langle \phi, \psi_1, \dots, \psi_m \rangle$ is called a **k - τ - σ - $\mathcal{L}$ -translation scheme** or, in short, a **translation scheme**, if the parameters are clear in the context.

Distinctions

If $k = 1$ we speak of **scalar** or **non-vectorized** translation schemes.

If $k \geq 2$ we speak of **vectorized** translation schemes.

If ϕ is such that $\forall \bar{x} \phi(\bar{x})$ is a tautology (always true) the translation scheme is **not relativized** otherwise it is **relativized**.

A translation scheme is **simple** if it is neither relativized nor vectorized.

Example 2 (τ_{words_3} and τ_{graphs})

τ_{words_3} consists of $\{R_<, P_0, P_1, P_2\}$ for three letters $\{0, 1, 2\}$.

τ_{graphs} consists of $\{E\}$

Put $k = 1$,

$\phi_1(x) = (P_0(x) \vee P_1(x))$ and

$\psi_E(x, y) = (P_0(x) \wedge P_1(y))$

$$\Phi_1 = \langle \phi_1(x), \psi_E(x, y) \rangle$$

is a **scalar** and **relativized** translation scheme in *FOL*.

If instead we look at $\phi_2(x) = (x \approx x)$ then

$$\Phi_2 = \langle \phi_2(x), \psi_E(x, y) \rangle$$

is a **simple** translation scheme.

Example 3 (τ_{words_2} and τ_{grids})

τ_{words_2} consists of $\{R_<, P_0, P_1\}$

τ_{grids} consists of $\{E_{NS}, E_{EW}\}$

Put $k = 2$,

$$\phi(x) = ((x \approx x) \wedge (y \approx y))$$

$$\psi_{E_{NS}}(x_1, x_2, y_1, y_2) = (R_<(x_1, x_2) \wedge y_1 \approx y_2)$$

$$\psi_{E_{EW}}(x_1, x_2, y_1, y_2) = (R_<(y_1, y_2) \wedge x_1 \approx x_2)$$

$$\Phi_3 =$$

$$\langle \phi(x, y), \psi_{E_{NS}}(x_1, x_2, y_1, y_2), \psi_{E_{EW}}(x_1, x_2, y_1, y_2) \rangle$$

is a **vectorized** but **not** relativized translation scheme in FOL.

Definition 4 (The induced transduction Φ^*)

Given a translation scheme Φ

$$\Phi^* : Str(\tau) \rightarrow Str(\sigma)$$

is a (partial) function from τ -structures to σ -structures defined by $\Phi^*(\mathcal{A}) = \mathcal{A}_\Phi$ and

(i) the universe of $\mathcal{A}_\Phi$ is the set

$$A_\Phi = \{\bar{a} \in A^k : \mathcal{A} \models \phi(\bar{a})\};$$

(ii) the interpretation of R_i in $\mathcal{A}_\Phi$ is the set

$$\mathcal{A}_\Phi(R_i) = \{\bar{a} \in A_\Phi^{\rho(R_i) \cdot k} : \mathcal{A} \models \psi_i(\bar{a})\}.$$

$\mathcal{A}_\Phi$ is a σ -structure of cardinality at most $|A|^k$.

As Φ is k -feasible for σ over τ , $\Phi^*(\mathcal{A})$ is defined iff $\mathcal{A} \models \exists \bar{x} \phi$.

Example 5 (Words and graphs)

Put $k = 1$, $\phi_1(x) = (P_0(x) \vee P_1(x))$ and $\psi_E(x, y) = (P_0(x) \wedge P_1(y))$

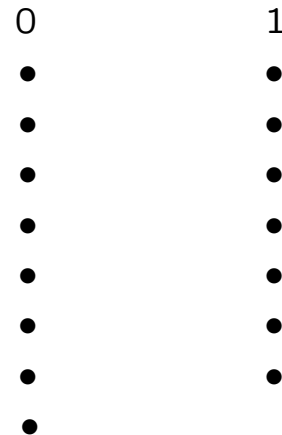
$$\Phi_1 = \langle \phi_1(x), \psi_E(x, y) \rangle$$

is a **scalar** and **relativized** translation scheme in *FOL*. Let us compute Φ_1^* .

For the word

1001020102001022111

we get the graph (**complete the edges**):



Example 6 (Words and grids)

Put $k = 2$, $\phi(x) = ((x \approx x) \wedge (y \approx y))$, $\psi_{ENS}(x_1, x_2, y_1, y_2) = (R_<(x_1, x_2) \wedge y_1 \approx y_2)$ and $\psi_{EW}(x_1, x_2, y_1, y_2) = (R_<(y_1, y_2) \wedge x_1 \approx x_2)$.

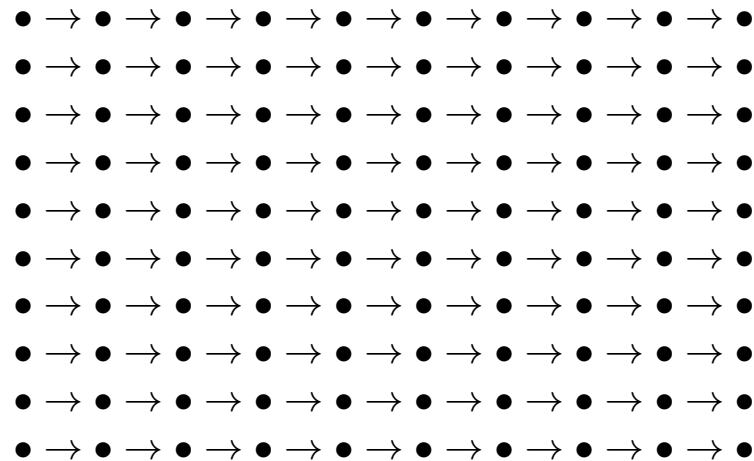
$$\Phi_3 = \langle \phi(x, y), \psi_{ENS}(x_1, x_2, y_1, y_2), \psi_{EW}(x_1, x_2, y_1, y_2) \rangle$$

is a **vectorized** but **not** relativized translation scheme in FOL. Let us compute Φ_3^* .

For a word

0110101001

we get the graph (**complete the edges**):



This is independent of the letters $\{0, 1\}$.

Definition 7 (The induced translation $\Phi^\#$)

Given a translation scheme Φ we define a function $\Phi^\# : \mathcal{L}(\sigma) \rightarrow \mathcal{L}(\tau)$ from $\mathcal{L}(\sigma)$ –formulae to $\mathcal{L}(\tau)$ –formulae inductively as follows:

- For $R_i \in \sigma$ and $\theta = R_i(x_1, \dots, x_m)$ let $x_{j,h}$ be new variables with $i \leq m$ and $h \leq k$ and denote by $\bar{x}_i = \langle x_{i,1}, \dots, x_{i,k} \rangle$. We put

$$\Phi^\#(\theta) = \left(\psi_i(\bar{x}_1, \dots, \bar{x}_m) \wedge \bigwedge_i \phi(\bar{x}_i) \right)$$

- This also works for equality and relation variables U instead of relation symbols R .

Definition 7 (Continued: booleans)

For the boolean connectives, the translation distributes, i.e.

- if $\theta = (\theta_1 \vee \theta_2)$ then

$$\Phi^\sharp(\theta) = (\Phi^\sharp(\theta_1) \vee \Phi_\sharp(\theta_2))$$

- if $\theta = \neg\theta_1$ then

$$\Phi_\sharp(\theta) = \Phi_\sharp(\neg\theta_1)$$

- similarly for $\wedge$ and $\rightarrow$.

Definition 7 (Continued: quantification)

- For the existential quantifier, we use relativization to ϕ :
If $\theta = \exists y \theta_1$, let $\bar{y} = \langle y_1, \dots, y_k \rangle$ be new variables. We put

$$\theta_\Phi = \exists \bar{y} (\phi(\bar{y}) \wedge (\theta_1)_\Phi).$$

This concludes the inductive definition for first order logic *FOL*.

- For second order quantification of variables U of arity ℓ and $\bar{a}$ a vector of length ℓ of first order variables or constants, we translate $U(\bar{a})$ by treating U as a relation symbol above and put

$$\theta_\Phi = \exists V (\forall \bar{v} (V(\bar{v}) \rightarrow (\phi(\bar{v}_1) \wedge \dots \wedge \phi(\bar{v}_\ell) \wedge (\theta_1)_\Phi)))$$

Example 8 (Computing $\Phi_1^\sharp$)

Recall

$$\Phi_1 = \langle \phi_1(x), \psi_E(x, y) \rangle$$

with $k = 1$,

$\phi_1(x) = (P_0(x) \vee P_1(x))$ and

$\psi_E(x, y) = (P_0(x) \wedge P_1(y))$

Let θ_{conn} be the formula which says the graph is connected:

$$\neg (\exists U (\exists x \neg U(x) \wedge \forall x \forall y (U(x) \wedge E(x, y) \rightarrow U(y))))$$

Example 8 (Continued)

- $U(x)$ is replaced by

$$(\phi_1(x) \wedge U(x)) = ((P_0(x) \vee P_1(x)) \wedge U(x))$$

- $E(x, y)$ is replaced by

$$\begin{aligned} (\phi_1(x) \wedge \phi_1(y) \wedge E(x, y)) = \\ ((P_0(x) \vee P_1(x)) \wedge (P_0(y) \vee P_1(y)) \wedge E(x, y)) \end{aligned}$$

- $(x \approx y)$ is replaced by

$$\begin{aligned} (\phi_1(x) \wedge \phi_1(y) \wedge (x \approx y)) = \\ ((P_0(x) \vee P_1(x)) \wedge (P_0(y) \vee P_1(y)) \wedge (x \approx y)) \end{aligned}$$

- Then we proceed inductively.

$(x \approx y)$ does not occur in θ_{conn} .

Fundamental Property of translation schemes

Theorem 9

Let

$$\Phi = \langle \phi, \psi_1, \dots, \psi_m \rangle$$

be a $k-(\tau - \sigma)$ -translation scheme in a logic $\mathcal{L}$, and $\mathcal{A}$ be a τ -structure and θ be a $\mathcal{L}(\sigma)$ -formula. Then

$$\boxed{\mathcal{A} \models \Phi^\sharp(\theta) \text{ iff } \Phi^*(\mathcal{A}) \models \theta}$$

Translation Scheme and its induced maps

in the Fundamental Property of theorem 9

Translation scheme Φ		
τ -structure $\mathcal{A}$	$\xrightarrow{\Phi^*}$	σ -structure $\Phi^*(\mathcal{A})$
τ -formulae $\Phi^\sharp(\theta)$	$\xleftarrow{\Phi^\sharp}$	σ -formulae θ
$\mathcal{A} \models \Phi^\sharp(\theta)$	iff	$\Phi^*(\mathcal{A}) \models \theta$

Quantifier rank of $\Phi^\#(\theta)$

Proposition 10

Let Φ be a translation scheme with all the formulas of quantifier rank $\leq q$, and θ a formula of quantifier rank $\leq r$.

Assume $\mathcal{A}$ and $\mathcal{B}$ are τ -structures and $\mathcal{A}, \mathcal{B}$ satisfy the same formulas of $\mathcal{L}(\tau)$ of quantifier rank $\leq q + r$.

(i) $\Phi^\#(\theta)$ has quantifier rank $\leq q + r$.

(ii) $\Phi^(\mathcal{A})$ and $\Phi^*(\mathcal{B})$ satisfy the same formulas of $\mathcal{L}(\sigma)$ of quantifier rank $\leq r$.*

Definition 11 ($\mathcal{L}$ –Reductions)

Let $\mathcal{L}$ be a regular logic and Φ be a $(\tau_1 - \tau_2)$ translation scheme. We are given

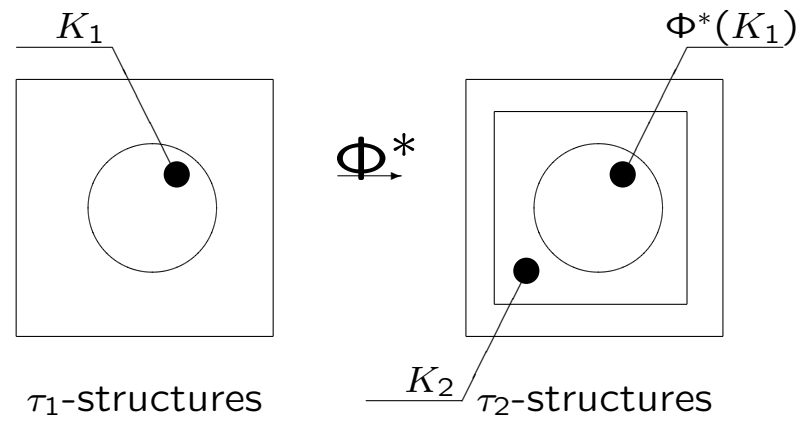
- two classes K_1, K_2 of $\tau_1(\tau_2)$ -structures closed under isomorphism

We say

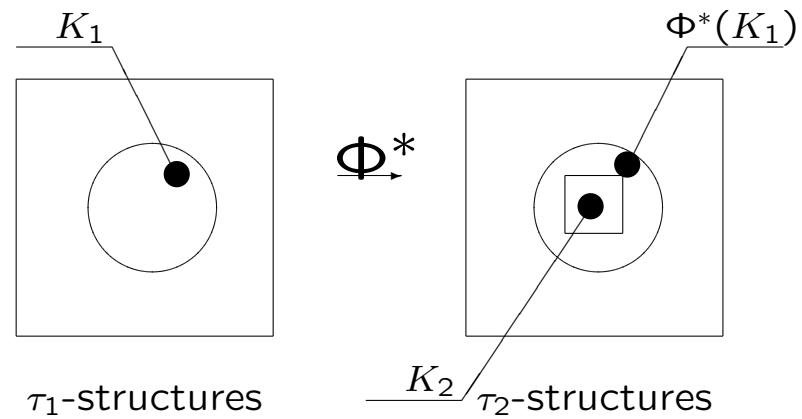
- (i) Φ^* is a **weak reduction** of K_1 to K_2 if for every τ_1 -structure $\mathfrak{A}$ with $\mathfrak{A} \in K_1$ we have $\Phi^*(\mathfrak{A}) \in K_2$.
- (ii) Φ^* is a **reduction** of K_1 to K_2 if for every τ_1 -structure $\mathfrak{A}$, $\mathfrak{A} \in K_1$ iff $\Phi^*(\mathfrak{A}) \in K_2$.

Definition 11(Continued)

- (iii) Φ^* of K_1 to K_2 is **onto** if (additionally) for every $\mathfrak{B} \in K_2$ there is an $\mathfrak{A} \in K_1$ with $\Phi^*(\mathfrak{A})$ isomorphic to $\mathfrak{B}$.
- (iv) By abuse of language we say Φ^* is a **translation of K_1 onto K_2** also if Φ^* is not a weak reduction but only $K_2 \subseteq \Phi^*(K_1)$.
- (v) We say that Φ induces a reduction (a weak reduction) of K_1 to K_2 , if Φ^* is a reduction (a weak reduction) of K_1 to K_2 . For simplicity, we also say Φ is a reduction (a weak reduction) instead of saying that Φ induces a reduction (a weak reduction).



Weak reduction



ONTO

Definition 12 ($\mathcal{L}$ -Reducibility)

- (i) Let $k \in \mathbb{N}$.
We say that K_1 is $\mathcal{L}$ - k -reducible to K_2 ($K_1 \triangleleft_{\mathcal{L}-k} K_2$), if there is a $\mathcal{L}$ - k -translation scheme Φ for τ_2 over τ_1 , such that Φ^* is a reduction of K_1 to K_2 .
- (ii) We say that K_1 is $\mathcal{L}$ -reducible to K_2 ($K_1 \triangleleft_{\mathcal{L}} K_2$), if $K_1 \triangleleft_{\mathcal{L}-k} K_2$ for some $k \in \mathbb{N}$.
- (iii) We say that K_1 is $\mathcal{L}$ -bi-reducible to K_2 and write $K_1 \bowtie_{\mathcal{L}} K_2$, if $K_1 \triangleleft_{\mathcal{L}-k} K_2$ and $K_2 \triangleleft_{\mathcal{L}-k} K_1$ for some $k \in \mathbb{N}$.
Clearly, bi-reducibility is a symmetric relation.

Theorem 13 (Definability and Reducibility)

Let Φ^* be an $\mathcal{L}$ -reduction of K_1 to K_2 .
If K_2 is $\mathcal{L}$ -definable then K_1 is $\mathcal{L}$ -definable.

Recall that a class of τ -structures K_2 is $\mathcal{L}$ -definable if there is a $\mathcal{L}(\tau)$ -sentence θ such that $K_2 = \text{Mod}(\theta)$.

Proof:

We use the Fundamental Property of Φ .

If K_2 is defined by θ , so K_1 is defined by $\Phi^\#(\theta)$.

Proposition 14

Hamiltonian graphs are not MSOL-definable in τ_{graph} but definable $\tau_{hgraphs}$.

Proof:

We use Φ_2 from example 2.

Φ_2^* is a reduction from words $0^n 1^m$ over $\{0, 1\}$ to complete bipartite graphs $K_{n,m}$, which are MSOL-defined by θ_{co-bi} .

$K_{n,m}$ is Hamiltonian iff $n = m$.

So, if θ_{hamil} defined all Hamiltonian graphs,

$$\Phi_2^\sharp(\theta_{hamil} \wedge \theta_{co-bi})$$

defined the language $\{0^n 1^n\}$.

But $\{0^n 1^n\}$ is not regular, and hence, by Büchi's theorem, not MSOL-definable.

For $\tau_{hgraphs}$ we just say that there is a subset of $E(G)$ which covers all vertices, is connected and regular of degree 2. Q.E.D.

Proposition 15

Eulerian graphs are not MSOL-definable both in τ_{graph} and $\tau_{hgraphs}$.

Proof: Let SET be the class of finite sets and $ODD \subseteq SET$ those of odd cardinality. Let $CLIQUE$ be the class of complete graphs. $CLIQUE$ is FOL -definable by some θ_{clique} .

Let the simple FOL translation scheme Φ be given by $\phi(x) = (x \approx x)$ and $\psi_E(x, y) = (\neg x \approx y)$.

Φ^* is a reduction from SET to $CLIQUE$.

Now assume that there is $\theta_{euler} \in MSOL$, with $EULER = Mod(\theta_{euler})$.

Put $\theta = (\theta_{clique} \wedge \theta_{euler})$.

$\Phi^\sharp(\theta)$ is equivalent to $\theta_{odd} \in MSOL$.

But this contradicts the fact that ODD ($EVEN$) is not $MSOL$ -definable.

For $\tau_{hgraphs}$ we leave it as an exercise.

Q.E.D.

Proof of theorem 9

We use induction over the construction of θ .

- If all the formulas ϕ, ψ_i of Φ and θ are atomic, both $\Phi^*(\mathfrak{A}) = \mathfrak{A}$ and $\Phi^\sharp(\theta) = \theta$.

- Next we keep θ atomic and assume

$$\Phi = \langle \phi(\bar{x}), \psi_{S_1}(\bar{x}), \dots, \psi_{S_m}(\bar{x}) \rangle$$

$\Phi^*(\mathfrak{A}) \models S_i(\bar{a})$ iff $\mathfrak{A} \models \psi_{S_i}(\bar{a})$
by definition of Φ^* .

- Now the induction on θ uses that $\Phi^\sharp$ commutes with the logical constructs.

Q.E.D.

Example 16 (Renaming)

One of the simplest translations encountered in logic is the renaming of basic relations.

Let $\tau_1 = \{R_i : i \leq k\}$ and $\tau_2 = \{S_i : i \leq k\}$, where R_i and S_i are of the same arity, respectively.

Let Φ be the (τ_1, τ_2) translation scheme given by $\Phi = \langle x = x, R_1(\bar{u}), \dots, R_k(\bar{v}) \rangle$.

Such a translation scheme and as well as its induced maps Φ^* and $\Phi^\sharp$ are called **renaming**.

Example 17 (Cartesian Product)

Let us consider one example of vectorized translation scheme that defines Cartesian Product.

For simplicity, we assume that $k = 2$.

Let $\tau_1 = \{R_1(x_1, x_2)\}$ with R_1 binary
and $\tau_2 = \{R_2(x_1, x_2)\}$ with R_2 binary.

$$\Phi = \langle (x_1 = x_1 \vee x_2 = x_2), \\ (R_1(x_1, x_2) \wedge R_2(x_3, x_4)) \rangle$$

It is easy to see that $\Phi^*(\mathcal{A})$ is isomorphic to the Cartesian product $\mathcal{A}^2$.

The n -hold Cartesian product is defined in the same way.

Example 18 (Graphs)

$Graphs_1$ is the class of structures of the form $\langle V, E \rangle$ where E is a binary ir-reflexiv relation on the set of vertices V .

$Graphs_2$ is the class of structures of the form $\langle V \sqcup E; Src(v, e), Tgt(v, e) \rangle$ with the universe consisting of **disjoint** sets of vertices and edges and $Src(v, e)$ ($Tgt(v, e)$) indicates that v is the source (target) of the directed edge e .

For a graph G we denote its representations by G_i for $G_i \in Graphs_i$ respectively.

We define a scalar translation scheme

$\Phi = \langle \phi, \psi_E \rangle$ from $Graphs_2$ to $Graphs_1$ by

$$\begin{aligned}\phi(v) &= (\exists e (Src(v, e) \vee e Tgt(v, e))) \vee \\ &\quad (v = v \wedge \neg \exists x (Src(x, v) \vee Tgt(x, v))) \\ \phi_E(x, y) &= \exists e ((Src(x, e) \wedge Tgt(y, e)))\end{aligned}$$

Clearly, for every graph G we have

$$\Phi^*(G_2) \cong G_1$$

Theorem 19 (Model checking for transductions)

If Φ is in *FOL* (or $\exists$ *HornSOL*)
then Φ^* is computable in polynomial time.

Proof:

We test all k -tuples $\bar{a}$ in $\mathfrak{A}$ of size n for

$$\mathfrak{A} \models \phi(\bar{a})$$

This takes $n^k \cdot \text{TIME}(\mathfrak{A}, \phi)$ time.

But we know that $\text{TIME}(\mathfrak{A}, \phi)$ is a polynomial in n .

For the ψ_{S_i} this is the same.

Q.E.D.

By a theorem of Grädel, this also holds for *HornESOL*, as we have shown in Part 2.

Part 4.2: The Feferman-Vaught Theorem

and its algorithmic uses

Based originally on my
Lecture originally prepared for the Tarski Centennial Conference
Warsaw, Poland, May, 2001

Vocabularies, structures and theories

Let τ vocabulary

(or a *similarity type* as Tarski used to call it)

given by a set of relation symbols, but no function symbols nor constants.

$\text{FOL}(\tau)$ denotes the set of τ -formulas in First Order Logic.

$\text{SOL}(\tau)$ and $\text{MSOL}(\tau)$ denote the set of τ -formulas in Second Order and Monadic Second Order Logic.

For a class of τ -structures K

$\text{Th}_{\text{FOL}}(K)$ is the set of sentences of $\text{FOL}(\tau)$ true in all $\mathfrak{A} \in K$.

We write $\text{Th}_{\text{FOL}}(\mathfrak{A})$ for $K = \{\mathfrak{A}\}$.

Similarly, $\text{Th}_{\text{SOL}}(K)$ and $\text{Th}_{\text{MSOL}}(K)$ for SOL and MSOL.

A. Tarski and E.W. Beth

Tarski published four short abstracts on model theory in 1949 (in the Bulletin of the AMS, vol. 55) and sent his manuscript for

Contribution to the theory of models, I

to E.W. Beth for publication.



Alfred Tarski (1901-1983)



Evert Willem Beth (1908-1964)

Inspired by these, E.W. Beth published two papers on model theory. In one of them he showed that

Beth's Theorem

Let $\mathfrak{A} = \langle A <_A \rangle$ and let $\mathfrak{B} = \langle B <_B \rangle$ be two linear orders.

We denote by $\mathfrak{C} = \mathfrak{A} \sqcup_{<} \mathfrak{B}$ their **ordered disjoint union**, defined by $C = A \sqcup B$, the disjoint union of A and B , and

$$<_C = <_A \sqcup B \cup A \times B$$

Theorem:(Beth 1952)

$Th_{\text{FOL}}(\mathfrak{C})$ is uniquely determined by $Th_{\text{FOL}}(\mathfrak{A})$ and $Th_{\text{FOL}}(\mathfrak{B})$.

In other words:

If $Th_{\text{FOL}}(\mathfrak{A}) = Th_{\text{FOL}}(\mathfrak{A}')$ and $Th_{\text{FOL}}(\mathfrak{B}) = Th_{\text{FOL}}(\mathfrak{B}')$ and $\mathfrak{C}' = \mathfrak{A}' \sqcup_{<} \mathfrak{B}'$ then $Th_{\text{FOL}}(\mathfrak{C}) = Th_{\text{FOL}}(\mathfrak{C}')$

In Tarski's school it was asked in the early 1950ties

Let $\mathfrak{A}, \mathfrak{B}$ be τ -structures,
 $\mathfrak{A} \times \mathfrak{B}$ their Cartesian product and
 $\mathfrak{A} \sqcup \mathfrak{B}$ their disjoint union.

Assume we are given $Th_{\text{FOL}}(\mathfrak{A})$ and $Th_{\text{FOL}}(\mathfrak{B})$.

What can we say about $Th_{\text{FOL}}(\mathfrak{A} \times \mathfrak{B})$ and $Th_{\text{FOL}}(\mathfrak{A} \sqcup \mathfrak{B})$?

What happens in the case of infinite sums and products?

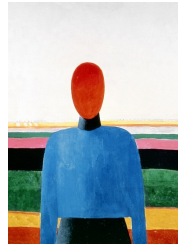
This triggered many landmark papers.

It also lead to the study of ultraproducts.

Tarski's pupils dealing with this question



A. Mostowski



A. Morel



S. Feferman



R. Vaught



J. Keisler

1938 Andrzej Mostowski

A. Mostowski, On direct product theories, JSL 17 (1952), pp. 1-31

1952 Anne Morel

T.E. Frayne, A.C. Morel and D.S. Scott, Reduced direct products, Fundamenta Mathematicae 51 (1962), pp.195-228

1954 Robert Vaught

1957 Solomon Feferman

S. Feferman and R.L. Vaught, The first order properties of algebraic systems, Fundamenta Mathematicae 47 (1959), pp. 57-103

1961 Jerome Keisler

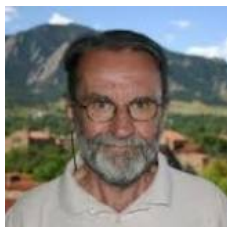
Many papers exploiting ultraproducts

Feferman and Vaught answered

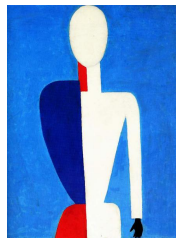
Theorem A:(Feferman and Vaught, 1959)

$Th_{\text{FOL}}(\mathfrak{A} \times \mathfrak{B})$ and $Th_{\text{FOL}}(\mathfrak{A} \sqcup \mathfrak{B})$ are uniquely determined by $Th_{\text{FOL}}(\mathfrak{A})$ and $Th_{\text{FOL}}(\mathfrak{B})$.

- By combining it with transductions and interpretations this remains true for a wide variety of generalized products.
- Also true for infinite generalized sums and products, provided the index structures are sufficiently MSOL indistinguishable.
- For MSOL still true for disjoint unions $\mathfrak{A} \sqcup \mathfrak{B}$ over infinite index sets, (Ehrenfeucht, Läuchli, Shelah, Gurevich), but **not true for SOL**.



A. Ehrenfeucht



H. Läuchli



S. Shelah



Y. Gurevich

Ehrenfeucht's proof theorem A

Use Ehrenfeucht-Fraïssé Games

This gives actually more:

Let $q \in \mathbb{N}$

and let $\text{FOL}^q(\tau)$ denote the sentences of FOL of quantifier rank at most q .

Put $Th_{\text{FOL}}^q(\mathfrak{A}) = Th_{\text{FOL}}(\mathfrak{A}) \cap \text{FOL}^q(\tau)$.

Theorem B: (Feferman and Vaught, 1959)

$Th_{\text{FOL}}^q(\mathfrak{A} \times \mathfrak{B})$ and $Th_{\text{FOL}}^q(\mathfrak{A} \sqcup \mathfrak{B})$
are uniquely determined by $Th_{\text{FOL}}^q(\mathfrak{A})$ and $Th_{\text{FOL}}^q(\mathfrak{B})$.

For MSOL still true for $\mathfrak{A} \sqcup \mathfrak{B}$.

Feferman and Vaught's proof

Use Reduction Sequences

Here one proves by induction, say for disjoint union

Theorem C:(Feferman and Vaught, 1959)

For every formula $\phi \in \text{FOL}^q(\tau)$ one can compute a sequence of formulas

$$\langle \psi_1^A, \dots, \psi_m^A, \psi_1^B, \dots, \psi_m^B \rangle \in \text{FOL}^q(\tau)^{2m}$$

and a Boolean function

$B_\phi : \{0, 1\}^{2m} \rightarrow \{0, 1\}$ such that

$$\mathfrak{A} \sqcup \mathfrak{B} \models \phi$$

iff

$$B_\phi(b_1^A, \dots, b_m^A, b_1^B, \dots, b_m^B) = 1$$

where $b_j^A = 1$ iff $\mathfrak{A} \models \psi_j^A$ and $b_j^B = 1$ iff $\mathfrak{B} \models \psi_j^B$

Similarly for MSOL

References

For a detailed proof of [Theorem C](#) constructing [reduction sequences](#), consult

- J.A. Makowsky
Algorithmic Uses of the Feferman-Vaught Theorem,
Annals of Pure and Applied Logic 126.1-3 (2004): 159-213.

We now proceed to give a proof using [Hintikka sentences](#).

From infinite to finite structures

- The Feferman-Vaught Theorems hold for infinite and finite structures.
- It plays an important role in model theory.
- It is usually not discussed in standard text books of logic.

Algorithmic problems on finite structures

From 1970 on interest in computing focused on computing on **finite structures**:

- **NP**-complete problems and complexity hierarchies.
- Graph algorithms and other finite data structures.
- Generalizing finite automata to trees and beyond.

In all these fields **Use Ehrenfeucht-Fraïssé Games** and the **Feferman-Vaught Theorem** play crucial rôles.

The Hintikka matrix for the (rich) disjoint union, I

We have already shown [Theorem B](#) using EF-games.

- There are only finitely many Hintikka sentences of quantifier rank q for a fixed relational vocabulary τ .
- Let $h_1^q, \dots, h_{\alpha_q}^q$ be an enumeration of all Hintikka sentences of quantifier rank q for a fixed relational vocabulary τ .
- By [Theorem B](#) and the properties of Hintikka sentences there is a function $f : [\alpha]^2 \rightarrow [\alpha]$ such that $\mathfrak{A} \models h_i^q$ and $\mathfrak{B} \models h_j^q$ iff $\mathfrak{A} \sqcup \mathfrak{B} \models h_{f(i,j)}^q$.
- f is not one-one: There structures over the empty vocabulary $[5] \sqcup [7]$ and $[4] \sqcup [8]$ satisfy the same Hintikka sentences.
- f is not onto: A connected graph cannot be the disjoint union of two graphs.
- The **Hintikka matrix** $H(\sqcup, q)$ is the $(\alpha \times \alpha)$ -matrix where the entry is given by $H(\sqcup, q)(i, j) = h_{f(i,j)}^q$.

The Hintikka matrix for the (rich) disjoint union, II

We now want to express the conclusion of [Theorem C](#) in terms of Hintikka sentences.

- Assume $\phi = h_k^q$ is a Hintikka sentence for $\mathfrak{A} \sqcup \mathfrak{B}$.
- Let $F_k = \{(i, j) \in [\alpha]^2 : f(i, j) = k\}$.
- The **reduction sequence** for ϕ is given by the two sequences
 $(h_i^q(\mathfrak{A}) : \mathfrak{A} \models h_i^q, \exists j f(i, j) = k)$ and $(h_j^q(\mathfrak{B}) : \mathfrak{B} \models h_j^q, \exists i f(i, j) = k)$
- The boolean function for ϕ is

$$\bigvee_{(i,j) \in F_k} (h_i^q(\mathfrak{A}) \wedge h_j^q(\mathfrak{B}))$$

The Hintikka matrix for the (rich) disjoint union, III

- If ϕ is not a Hintikka sentence, it still can be written as a disjunction

$$\bigvee_{i, h_i^q \rightarrow \phi} h_i^q$$

of Hintikka sentences.

- We now do the proof for each h_i^q and piece the reduction sequence and the boolean function together.

Complexity of finding the reduction sequence

- The proof using Hintikka formulas seems very **wasteful**.
It relies on computing **all the Hintikka sentences of quantifier rank q** .
- The recursive construction depends only on the structure of ϕ .
If ϕ is of **length n** , it requires only n iterations.
But the proof will yield a formula of length $tower(n, n)$.
Here $tower(1, n) = O(2^n)$, $tower(m + 1, n) = O(2^{tower(m, n)})$
- This is almost best possible:
Dawar, A., Grohe, M., Kreutzer, S., and Schweikardt, N.
Model theory makes formulas large.
In International Colloquium on Automata, Languages, and Programming
(2007). pp. 913-924. Springer, Berlin, Heidelberg. .
- In special cases, the reduction sequence can be **very short**.
A graph $G_1 \sqcup G_2$ is a forest iff both G_1 and G_2 are forests.

The Hankel matrix of a graph property with respect to the disjoint union $\sqcup$

- Herrmann Hankel was a German mathematician. He introduced Hankel matrices in **linear algebra**.
- Hankel matrices were generalized to be used in automata theory .
- Hankel matrices are called connection matrices in L. Lovasz' work on **Large Network and Graph Limits**

Hankel matrix $M(\mathcal{P}, \sqcup)$.

Let G_i be an enumeration of all finite graphs (up to isomorphism).

The (full) **Hankel matrix** $M(\mathcal{P}, \sqcup) = m_{i,j}(\mathcal{P}, \sqcup)$ is defined by

$$m_{i,j}(\mathcal{P}, \sqcup) = \begin{cases} 1 & G_i \sqcup G_j \in \mathcal{P} \\ 0 & \text{else} \end{cases}$$

The **rank** of $M(\mathcal{P}, \sqcup)$ is denoted by $r(\mathcal{P}, \sqcup)$.

We shall often look at various infinite submatrices of the full Hankel matrix.

Examples: Check with *3COL*, *CONN* and *PLANAR*.

General Connection Matrices (aka Hankel Matrices): I

Let $\mathcal{C}$ be a class possibly **labeled graphs**, **hyper-graphs** or **τ -structures**.

Let $\square$ be a binary operation defined on $\mathcal{C}$.

Let G_i be an enumeration of all (labeled) finite graphs (structures) in $\mathcal{C}$.

Let $\mathcal{P}$ be graph property.

The (full) connection matrix $M(\mathcal{P}, \square)$ is defined by

$$m_{i,j}(\mathcal{P}, \square) = \begin{cases} 1 & G_i \square G_j \in \mathcal{P} \\ 0 & \text{else} \end{cases}$$

and is called the **Full Hankel Matrix of f for $\square$ on $\mathcal{C}$** ,

or just a **Hankel matrix**.

We denote by $r(\mathcal{P}, \square)$ the **rank** of $M(\mathcal{P}, \square)$.

We shall often look at infinite submatrices of $M(\mathcal{P}, \square)$.

$\mathcal{L}$ -smooth operations.

Let $\mathcal{L}$ be a logic.

We say that two graphs G, H are $(\mathcal{L}, q)$ -equivalent, and write $G \sim_{\mathcal{L}}^q H$, if G and H satisfy the same $\mathcal{L}$ -sentences of quantifier rank q .

We say that $\square$ is $\mathcal{L}$ -smooth, if whenever we have

$$G_i \sim_{\mathcal{L}}^q H_i, i = 0, 1$$

then

$$G_0 \square G_1 \sim_{\mathcal{L}}^q H_0 \square H_1$$

This definition can be adapted to k -ary operations for $k \geq 1$.

Proving that an operation $\square$ is $\mathcal{L}$ -smooth may be difficult.

For FOL this can be achieved using

Ehrenfeucht-Fraïssé games also known as **pebble games**.

Another way of establishing smoothness is via the **Feferman-Vaught theorem**.

Examples of $\mathcal{L}$ -smooth operations.

- (i) Quantifier-free **scalar** transductions are both FOL and MSOL-smooth.
- (ii) Quantifier-free **vectorized** transductions are FOL but not MSOL-smooth.
- (iii) The **cartesian product** is FOL-smooth but not MSOL-smooth.
This was shown by **A. Mostowski in 1952**.
- (iv) The **(rich) disjoint union** is both FOL and MSOL-smooth.
The **rich** disjoint union has two additional unary predicates to distinguish the universes.
For FOL this was shown by **E. Beth in 1952**.
For MSOL this is due to H. Läuchli, 1966, using **Ehrenfeucht-Fraïssé games**.
- (v) **Adding modular counting quantifiers preserves smoothness.**
For CMSOL and the **disjoint union** this is due to **B. Courcelle, 1990**.
NEW: For *CFOL* and the **product** this is due to **T. Kotek and J.A.M., 2012**.

The Finite Rank Theorem

THEOREM (Godlin, Kotek, Makowsky 2008):

Let $\mathcal{P}$ be a property of τ -structures definable in $\text{MSOL}(\tau)$.

Let $\square$ be an MSOL-smooth operation.

Then the Hankel matrix $M(\mathcal{P}, \square)$ has **finite rank** over $\mathbb{Z}_2$.

The **Proof** uses a Feferman-Vaught-type theorem and Hintikka sentences.

Back to outline

Proof of the Finite Rank Theorem (outline)

Let $\mathcal{P}$ be a graph property definable by $\phi \in \text{MSOL}$ of quantifier rank q , and $\square$ be an MSOL-smooth operation.

Let $M(\mathcal{P}, \square)$ its Hankel matrix.

- For two graphs G, H we check $G \square H \in \mathcal{P}$. This depends only on the Hintikka sentences $h^q(G)$ and $h^q(H)$ of quantifier rank q .
- If two graphs G_1, G_2 satisfy the same Hintikka sentence h^k then the rows of $M(\mathcal{P}, \square)$ indexed by G_1 and G_2 are the same.
- There are only finitely many Hintikka sentences h^k , say $\alpha(k) \in \mathbb{N}$.
- Therefore the rank of the infinite matrix $M(\mathcal{P}, \square)$ is bounded by $\alpha(k)$.

Q.E.D.

Applications of the Finite Rank Theorem, I

Direct (categorical) products combined with translation schemes

The transduction

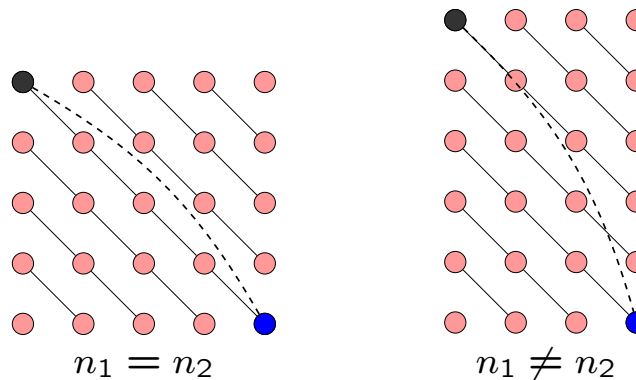
$$\Phi_F((v_1, v_2), (u_1, u_2)) = (E_1(v_1, u_1) \wedge E_2(v_2, u_2)) \vee ((v_1, v_2), (u_1, u_2)) = ((start_1, start_2), (end_1, end_2))$$

transforms the direct product of two **directed paths** $P_{n_i}^i = (V_i, E_i, start_i, end_i)$ of length n_i with the two constants $start_i$ and end_i , $i = 1, 2$ into an undirected graph with at most one cycle.

The input graphs look like this:



The result of the transduction is:



THEOREM: Graphs without cycles of odd (even) length
are not CFOL-definable **even in the presence of a linear order.**

Corollary: Not definable in CFOL with order are

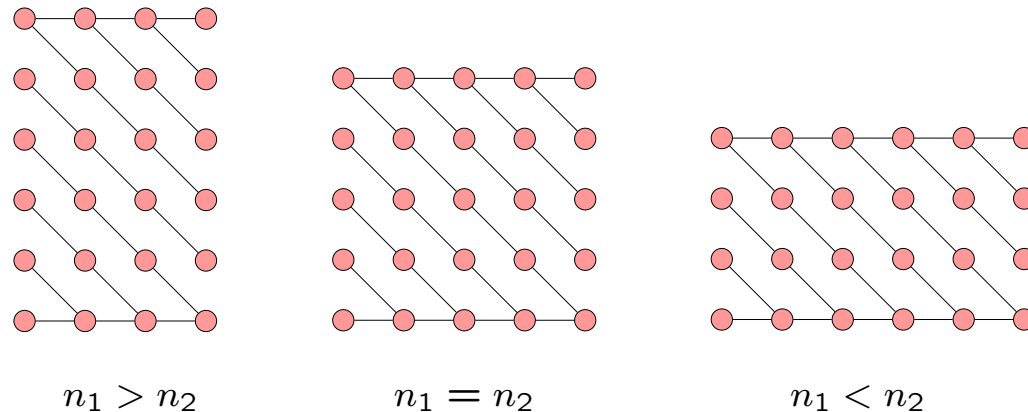
- (i) Forests, bipartite graphs, chordal graphs, perfect graphs
- (ii) interval graphs (cycles are not interval graphs)
- (iii) Block graphs (every biconnected component is a clique)
- (iv) Parity graphs (any two induced paths joining the same pair of vertices have the same parity)

THEOREM: Trees or connected graphs are not CFOL-definable even in the presence of linear order.

The transduction

$$\begin{aligned} \Phi_T((v_1, v_2), (u_1, u_2)) = & (E_1(v_1, u_1) \wedge E_2(v_2, u_2)) \vee \\ & (v_1 = u_1 = \text{start}_1 \wedge E(v_2, u_2)) \vee \\ & (v_1 = u_1 = \text{end}_1 \wedge E(v_2, u_2)), \end{aligned}$$

combined with Φ_{sym} transforms the cartesian product of two **directed paths** into the structures below:



Tree: $n_1 = n_2$. Connected: $n_1 \leq n_2$

k -graphs and k -sums

A k -graph is a graph $G = (V(G), E(G))$

with k distinct vertices labeled with $0, 1, \dots, k - 1$.

Given two k -graphs G_1, G_2 we define the k -sum

$$G_1 \sqcup_k G_2$$

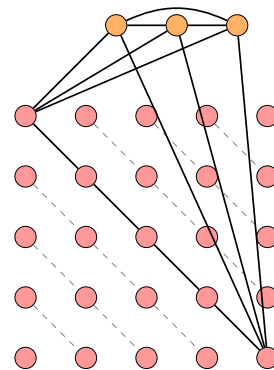
as the disjoint union of G_1 and G_2 where we
identify correspondingly labeled vertices.

Theorem: The k -sum is smooth for FOL, CFOL, MSOL and CMSOL.

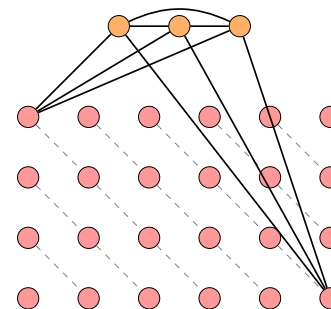
THEOREM: Planar graphs are not CFOL-definable even on ordered connected graphs

For our next connection matrix we use the 2-sum of the following two 2-graphs:

- the 2-graph (G, a, b) obtained from K_5 by choosing two vertices a and b and removing the edge between them
- the cartesian product of the two graphs $P_{n_1}^1$ and $P_{n_2}^2$:



$$n_1 = n_2$$



$$n_1 \neq n_2$$

The result of this construction has a clique of size 5 as a minor iff $n_1 = n_2$.

It can never have a $K_{3,3}$ as a minor.

Non-definability in CMSOL for graphs $G = (V, E)$

Using the join operation

The **join operation** of **graphs** $G = (V, E)$, where E is the **edge relation**, is defined by

$$(V_1, E_1) \bowtie (V_2, E_2) = (V_1 \sqcup V_2, E_1 \sqcup E_2 \cup \{(v_1, v_2) : v_1 \in V_1, v_2 \in V_2\})$$

This is a quantifier free transduction of the disjoint union, hence smooth for CMSOL.

Consider the connection matrix where the rows and columns are labeled by the graphs on n vertices but without edges E_n .

The graph $E_i \bowtie E_j = K_{i,j}$ is

- hamiltonian iff $i = j$;
- has a perfect matching iff $i = j$;
- is a cage graph (a regular graph with as few vertices as possible for its girth) iff $i = j$;
- is a well-covered graph (every minimal vertex cover has the same size as any other minimal vertex cover) iff $i = j$.

All of these connection matrices have infinite rank.

Proposition: None of the properties above are CMSOL-definable as graphs even in the presence of an order.

CMSOL for hyper-graphs $G = (V, E; R)$

A hyper-graphs $G = (V, E; R)$ has vertices V and edges E and an incidence relation R between the two.

- CMSOL for hyper-graphs $G = (V, E; R)$ allows quantification over edge sets.
- For the language of hypergraphs the join operation is neither MSOL- nor CMSOL-smooth, since it increases the number of edges.
- Note also that hamiltonicity and having a perfect matching are both definable in CMSOL in the language of hypergraphs.

In the many papers of B. Courcelle, MSOL on graphs is called MSOL_1 and for hyper-graphs it is called MSOL_2 .

Non-definability in CMSOL for hyper-graphs $G = (V, E; R)$

Using the disjoint union

Using the connection submatrices of the disjoint union we still get:

- Regular: $K_i \sqcup K_j$ is regular iff $i = j$;
- A generalization of regular graphs are *bidegree* graphs, i.e., graphs where every vertex has one of two possible degrees. $K_i \sqcup (K_j \sqcup K_1)$ is a bidegree graph iff $i = j$.
- A graph is *asymmetric* (or *rigid*) if it has no non-trivial automorphisms. It was shown by P. Erdős and A. Rényi (1963) that almost all finite graphs are asymmetric. So there is an infinite set $I \subseteq \mathbb{N}$ such that for $i \in I$ there is an asymmetric graph R_i of cardinality i . $R_i \sqcup R_j$ is asymmetric iff $i \neq j$.

Proposition: None of the properties above are CMSOL-definable as hyper-graphs even in the presence of an order.

Back to outline

End of session 7 (December 16, 2021)

Session 8 (December 23, 2021)

Operations on coloured, (un)-directed graphs.

Instead for the disjoint union we can prove theorem B and C for the following operations:

- Concatenation of words, $v \circ w$.
- Joining two trees
at a new common root, $T_1 \bullet T_2$.
- H -sums of graphs: For $i = 1, 2$
let $G_i = \langle V(G_i), E(G_i) \rangle$ and
 $V(G_1) \cap V(G_2) = V(H)$ and
 $E(H) = E(G_1) \cap V(H)^2 = E(G_2) \cap V(H)^2$.

Then $G = G_1 \oplus_H G_2$ is given by
 $V(G) = V(G_1) \cup V(G_2)$ and
 $E(G) = E(G_1) \cup E(G_2)$.

and similarly for edge and vertex coloured graphs

MSOL-smooth operations

We generalize the previous operations to operations satisfying theorem B or C:

Definition:

A n -ary operation $\mathcal{O}$ on τ -structures is **MSOL-smooth** if for every $q \in \mathbb{N}$ and every $\mathfrak{A}_1, \mathfrak{A}_2, \dots, \mathfrak{A}_n$

$$Th_{\text{MSOL}}^q(\mathcal{O}(\mathfrak{A}_1, \dots, \mathfrak{A}_n))$$

depends only on $Th_{\text{MSOL}}^q(\mathfrak{A}_i)$ for $1 \leq i \leq n$.

$\mathcal{O}$ is **effectively MSOL-smooth** if there is an algorithm which computes for every $\phi \in \text{MSOL}(\tau)$ a **reduction sequence**, i.e. a sequence of formulas as described in theorem C.

Examples of MSOL-smooth operations

- The disjoint union is effectively MSOL-smooth.
- Quantifier free **MSOL-transductions** are effectively MSOL-smooth.
- The **fusion operation** fus_P
i.e. identifying all elements which satisfy
a given unary predicate P
is MSOL-smooth, but we don't know whether it is effectively so.

Open problem:

Are there more examples?

Are there MSOL-smooth operations
which are not effectively MSOL-smooth ?

MSOL-inductive classes (graph grammars)

Definition:

A class K of τ -structures is **MSOL-inductive** if it is defined inductively using a finite set of MSOL-smooth operations.

K is **effectively MSOL-inductive** if it is defined inductively using a finite set of effectively MSOL-smooth operations.

Open problem:

Are there MSOL-inductive classes K which are not effectively MSOL-inductive ?

Effectively MSOL-inductive classes of structures, I

- Words Σ^* are defined inductively by
 - (i) the empty word is a word
 - (ii) one letter words are words
 - (iii) words are closed under concatenation
- Coloured trees (forests) are defined similarly:
 - (i) one leaf trees are trees
 - (ii) trees are closed under root joining
 - (iii) forests are closed under disjoint unions
- Series-parallel (SP) graphs are defined by
 - (i) one edge graphs are SP.
 - (ii) SP graphs are closed under disjoint unions
 - (iii) SP graphs are closed under H -sums for all H with at most two vertices.

Effectively MSOL-inductive classes of structures, II

- Series-parallel (SP) graphs are defined by
 - (i) one edge graphs are SP.
 - (ii) SP graphs are closed under disjoint unions
 - (iii) SP graphs are closed under H -sums for all H with at most two vertices.
- Co-graphs are define by
 - (i) one edge graphs are Co-graphs.
 - (ii) Co-graphs are closed under disjoint unions
 - (iii) Co-graphs are closed under the join operation.

More effectively MSOL-inductive classes

- Graphs of **tree width at most k** TW_k can be defined inductively by looking at vertex coloured graphs with at most $k + 1$ colours:
 - (i) All graphs with at most $k + 1$ vertices are in TW_k .
 - (ii) TW_k is closed under disjoint union.
 - (iii) TW_k is closed under renaming of colours.
 - (iv) TW_k is closed under **fusion**, i.e. contraction of all vertices of a specific colour into one vertex.
- Similarly, for graphs of **clique width at most k** CW_k
 - (i) All graphs with at most 1 vertex are in CW_k .
 - (ii) CW_k is closed under disjoint union.
 - (iii) CW_k is closed under renaming of colours.
 - (iv) CW_k is closed under adding all possible edges between to sets of differently coloured edges.

Representation of structures in MSOL-inductive classes

We can represent the structure $\mathfrak{A}$
by its relational table
or by a parse term $t_{\mathfrak{A}}$
which displays why it is in K .

In general, finding $t_{\mathfrak{A}}$ is **NP**-complete.

Theorem E:(Courcelle and M., 2001)

Let K be an CMSOL-inductive class of τ -structures and $\phi \in \text{CMSOL}(\tau)$.

Given a parse term $t_{\mathfrak{A}}$ for $\mathfrak{A}$, then the problem of deciding

$$\mathfrak{A} \models \phi$$

can be decided in linear time (in the size of $t_{\mathfrak{A}}$).

Model checking, I

Model checking is the problem to check

$$\mathfrak{A} \models \phi$$

for $\mathfrak{A}$ a finite τ structure and $\phi \in \text{SOL}(\tau)$.

We measure the problem in the size of $\mathfrak{A}$ and ϕ (combined case) or for specific ϕ .

Theorem:(M. and Mariño, 1996)

Even for MSOL there are ϕ such that the problem is arbitrarily high in the polynomial hierarchy.

Theorem:(Vardi, 1982)

The combined problem is **PSpace-complete** for SOL and even for FOL.

Proving these Theorem can be a suitable project!

Model checking, II

We want to do model checking on MSOL-inductive classes K .

We can represent the structure $\mathfrak{A}$ by its **relational table** or by a **parse term** $t_{\mathfrak{A}}$ which displays why it is in K .

In general, finding $t_{\mathfrak{A}}$ is **NP**-complete.

Theorem E:(Courcelle and M., 2001)

Let K be an MSOL-inductive class of τ -structures and $\phi \in \text{MSOL}(\tau)$.

Given a parse term $t_{\mathfrak{A}}$ for $\mathfrak{A}$, then the problem of deciding

$$\mathfrak{A} \models \phi$$

can be decided in linear time (in the size of $t_{\mathfrak{A}}$).

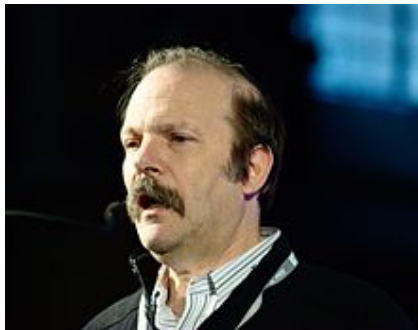
Proof of theorem E

$\text{MSOL}^q(\tau)$ is finite for every finite relational τ .

If all the operations are effectively MSOL-smooth, there is an algorithm which computes for every ϕ the look-up table (using theorem C).

Otherwise, we don't have such an algorithm, but still for each ϕ the look-up table for complete $\text{MSOL}^q(\tau)$ -types is finite (using theorem B).

Now we can compute along $t_{\mathfrak{A}}$, in the style of *dynamic programming*.



Moshe Vardi



Bruno Courcelle



Julian Mariño

Hankel matrices for graph parameters

From graph properties to graph parameters

- A **graph parameter** is a function f which maps graphs into a ring $\mathcal{R}$ and is invariant under graph isomorphisms.
- We can adapt the definition to functions f which map **relational τ -structures** graphs into a ring $\mathcal{R}$ and is invariant under τ -structures.
- The ring $\mathcal{R}$ in many considered cases is $\mathbb{Z}$, $\mathbb{Q}$ or $\mathbb{R}$.
- If $\mathbb{R}$ is a polynomial ring, say $\mathbb{Z}[\bar{X}]$ or $\mathbb{R}[\bar{X}]$, we call $f(G; \bar{X})$ a **graph polynomial**.

Motivation and History of Hankel matrices

Skip motivation and history

CMSOL-definable graph parameters

- I have developed with various co-authors a framework of **definability** of numeric graph parameters.
B. Courcelle, B. Godlin, T. Kotek, E. Ravve
- In this lecture we discuss a method of proving **non-definability** in **Monadic Second Order Logic with modular Counting CMSOL** of numeric graph parameters which take values in a field.
- The CMSOL-definable **graph parameters** behave similarly like CMSOL-definable **graph properties**.
 - (i) On graphs of **bounded width** they are in **FPT**, where the notion of width and the the notion of monadic quantification have to fit correspondingly.
 - (ii) All **classical graph polynomials** (Tutte polynomial, matching polynomial, chromatic polynomial, interlace polynomial) and many more are CMSOL-definable **using order on vertices in an invariant way**
 - (iii) On recursively defined graph sequences (like P_n , C_n , L_n , etc) they can be computed via linear recurrence relations.

Hankel aka connection matrices

Hankel matrices (over a ring $\mathcal{R}$)

Let $f : \mathcal{R} \rightarrow \mathcal{R}$ be a function over a ring $\mathcal{R}$.

A **finite** or **infinite** matrix $H(f) = h_{i,j}$ is a **Hankel matrix for f** if $H_{i,j} = f(i+j)$.

Hankel matrices have **many** applications in:
numeric analysis, **probability theory** and **combinatorics**.

- Padé approximations
- Orthogonal polynomials
- Probability theory (theory of moments)
- Coding theory (BCH codes, Berlekamp-Massey algorithm)
- Combinatorial enumerations
(Lattice paths, Young tableaux, matching theory)

Hankel matrices over words

Let Σ be a finite alphabet and $\mathcal{F}$ be a field and let $f : \Sigma^* \rightarrow \mathcal{F}$ be a function on words.

A **finite** or **infinite** matrix $H(f) = h_{u,v}$ indexed over the words $u, v \in \Sigma^*$ is a **Hankel matrix for f** if $h_{u,v} = f(u \circ v)$. Here $\circ$ denotes concatenation.

Hankel matrices over words have applications in

- Formal language theory and stochastic automata,
J. Carlyle and A. Paz 1971
- Learning theory (exact learning of queries).
A. Beimel, F. Bergadano, N. Bshouty, E. Kushilevitz, S. Varricchio 1998
J. Oncina 2008
- Definability of picture languages.
O. Matz 1998, and D. Giammarresi and A. Restivo 2008

Multiplicity Automata

A Multiplicity Automaton (MA) A of size r is given by:

- A set $\{\mu_\sigma : \sigma \in \Sigma\}$ of $r \times r$ matrices over $\mathcal{F}$;
- Two vectors $\lambda, \gamma \in \mathcal{F}^r$.
- A defines a function $f_A : \Sigma^* \rightarrow \mathcal{F}$

$$f_A(w) = f_A(\sigma_1 \circ \sigma_2 \circ \dots \circ \sigma_n) = \lambda \mu_{\sigma_1} \mu_{\sigma_2} \cdot \dots \cdot \mu_{\sigma_n} \gamma^t$$

THEOREM: (J. Carlyle and A. Paz 1971)

A function $f : \Sigma^* \rightarrow \mathcal{F}$ is representable as f_A for some MA A iff the Hankel matrix $H(f)$ has finite rank.

Hankel matrices for graphs

If we want to define Hankel matrices for (labeled) graphs,
what plays the role of concatenation?

- Disjoint union
Used by Freedman, Lovász and Schrijver, 2007, for
characterizing multiplicative graph parameters over the real numbers
- k -unions (connections, connection matrices)
Used by Freedman, Lovász, Schrijver and Szegedy, 2007ff, for
characterizing various forms and partition functions.
- Joins, cartesian products, generalized sum-like operations
used by Godlin, Kotek and JAM (2008) to prove non-definability.

Numeric graph invariants (graph parameters)

We denote by $G = (V(G), E(G))$ a graph,
and by $\mathcal{G}$ and $\mathcal{G}_{simple}$ the class of finite (simple) graphs, respectively.

A **numeric graph invariant** or **graph parameter** is a function

$$f : \mathcal{G} \rightarrow \mathbb{R}$$

which is invariant under graph isomorphism.

- (i) Cardinalities: $|V(G)|$, $|E(G)|$
- (ii) Counting configurations:
 - $k(G)$ the number of connected components,
 - $m_k(G)$ the number of k -matchings
- (iii) Size of configurations:
 - $\omega(G)$ the clique number
 - $\chi(G)$ the chromatic number

End of session 8 (December 23, 2021)

Begin of session 9 (December 30, 2021)

Definability of numeric graph parameters, I

We first give examples where we use **small**, i.e., polynomial sized sums and products:

(i) The cardinality of V is FOL-definable by

$$\sum_{v \in V} 1$$

(ii) The number of connected components of a graph G , $k(G)$ is MSOL-definable by

$$\sum_{C \subseteq V: \text{component}(C)} 1$$

where $\text{component}(C)$ says that C is a connected component.

(iii) The graph polynomial $X^{k(G)}$ is MSOL-definable by

$$\prod_{c \in V: \text{first-in-comp}(c)} X$$

if we have a linear order in the vertices and $\text{first-in-comp}(c)$ says that c is a first element in a connected component.

Definability of numeric graph parameters, II

Now we give examples with possibly **large**, i.e., exponential sized sums:

- (iv) The number of cliques in a graph is MSOL-definable by

$$\sum_{C \subseteq V: \text{clique}(C)} 1$$

where $\text{clique}(C)$ says that C induces a complete graph.

- (v) Similarly “the number of maximal cliques” is MSOL-definable by

$$\sum_{C \subseteq V: \text{maxclique}(C)} 1$$

where $\text{maxclique}(C)$ says that C induces a maximal complete graph.

- (vi) The number of cliques of maximal size of G is **SOL-definable** by

$$\sum_{C \subseteq V: \text{largest-clique}(C)} 1$$

where $\text{largest-clique}(C)$ says that C induces a maximal complete graph of largest size. We do need **full SOL** in order to express this.

Definability of numeric graph parameters, III

A numeric graph parameter is $\mathcal{L}$ -definable if it can be defined by similar expressions using **large** and **small sums** and only **small products**.

Usually, **summation** is allowed over **second order variables**, whereas **products** are over **first order variables**.

How can we prove **definability** and **non-definability** of graph parameters in some logic $\mathcal{L}$? In particular:

- How to prove that $k(G)$ is not CFOL-definable?
- How to prove that $\omega(G)$ is not CMSOL-definable?
- How to prove that the chromatic number $\chi(G)$ or the chromatic polynomial $\chi(G, X)$ is not CMSOL-definable?

Back to outline of Lecture 3

More definable graph parameters

- Let $m_k(G)$ be the number of sets isolated edges of G all of which have size k .

The **generating matching polynomial** is defined as

$$M(G; X) = \sum_k m_k(G) X^k = \sum_{F \subseteq E(G): \text{match}(F)} \prod_{e \in F} X$$

This is definable in MSOL for hypergraphs.

- Let $in_k(G)$ be the number of sets isolated vertices of G all of which have size k .

The **independence polynomial** is defined as

$$In(G; X) = \sum_k in_k(G) X^k = \sum_{U \subseteq V(G): \text{indep}(U)} \prod_{v \in U} X$$

This is definable in MSOL for graphs.

Theorem E for graph parameters

Theorem E:(Courcelle and M., 2001)

Let K be an CMSOL-inductive class of τ -structures and $\phi \in \text{CMSOL}(\tau)$.

Given a parse term $t_{\mathfrak{A}}$ for $\mathfrak{A}$, then computing an CMSOL-definable graph parameter $f(G)$ for graphs $G \in K$ of deciding

$$\mathfrak{A} \models \phi$$

can be done in linear time (in the size of the parse term t_G).

-

The Chromatic Polynomial

and

Its Variations

[Back to outline of lecture 2](#)

The (vertex) chromatic polynomial

Let $G = (V(G), E(G))$ be a graph, and $\lambda \in \mathbb{N}$.

A **λ -vertex-coloring** is a map

$$c : V(G) \rightarrow [\lambda]$$

such that $(u, v) \in E(G)$ implies that $c(u) \neq c(v)$.

We define $\chi(G, \lambda)$ to be the number of λ -vertex-colorings

Theorem: (G. Birkhoff, 1912)

$\chi(G, \lambda)$ is a polynomial in $\mathbb{Z}[\lambda]$.

Proof:

- (i) $\chi(E_n) = \lambda^n$ where E_n consists of n isolated vertices.
- (ii) For any edge $e \in E(G)$ we have $\chi(G - e, \lambda) = \chi(G, \lambda) - \chi(G/e, \lambda)$.

Interpretation of $\chi(G, \lambda)$ for $\lambda \notin \mathbb{N}$

What's the point in considering $\lambda \notin \mathbb{N}$?

Stanley, 1973 For simple graphs G , $|\chi(G, -1)|$ counts the number of **acyclic orientations** of G .

Stanley, 1973 There are also combinatorial interpretations of $\chi(G, -m)$ for each $m \in \mathbb{N}$, which are more complicated to state.

Open: What about $\chi(G, \lambda)$ for each $m \in \mathbb{R} - \mathbb{Z}$?

The Four Color Conjecture

Birkhoff wanted to prove the Four Color Conjecture using techniques from **real or complex analysis**.

Conjecture:(Birkhoff and Lewis) If G is planar then $\chi(G, \lambda) \neq 0$ for $\lambda \in [4, +\infty) \subseteq \mathbb{R}$.

This was not very successful. However, for **real roots** of χ we know:

Jackson, 1993 For simple graphs G we have $\chi(G, \lambda) \neq 0$ for $\lambda \in (-\infty, 0)$, $\lambda \in (0, 1)$ and $\lambda \in (1, \frac{32}{27})$.

Birkhoff and Lewis, 1946 For planar graphs G we have $\chi(G, \lambda) \neq 0$ for $\lambda \in [5, +\infty)$.

Still open: Are there planar graphs G such that $\chi(G, \lambda) = 0$ for some $\lambda \in (4, 5)$?

Thomassen, 1997 and Sokal, 2004 The real roots of all chromatic polynomials are dense in $(\frac{32}{27}, k]$ for graphs of tree-width at most k .
The complex roots are dense in $\mathbb{C}$.

Definability of the chromatic polynomial

Let $k \in \mathbb{N}$ be fixed.

The number of proper colorings of a graph with at most k colors is an MSOL-definable graph parameter.

$$\chi(G; k) = \sum_{U_1, \dots, U_k : (G, \bar{U}) \models \phi(\bar{U})} 1$$

where $\phi(\bar{U})$ says that $U_1, \dots, U_k$ partitions $V(G)$ into independent sets.

It is not obvious whether the polynomial $\chi(G; x)$ is MSOL-definable.

Variations on coloring, I

We can count other coloring functions.

- **proper λ -edge-colorings:**

$f_E : E(G) \rightarrow [\lambda]$ such that if $(e, f) \in E(G)$ have a common vertex then $f_E(e) \neq f_E(f)$.

$\chi_e(G, \lambda)$ denotes the number of λ - edge-colorings

- **Total colorings**

$f_V : V \rightarrow [\lambda_V]$, $f_E : E \rightarrow [\lambda_E]$ and $f = f_V \cup f_E$,

with f_V a proper vertex coloring and f_E a proper edge coloring.

- **Connected components**

$f_V : V \rightarrow [\lambda_V]$, If $(u, v) \in E$ then $f_V(u) \neq f_V(v)$.

- **Hypergraph colorings**

Vitaly I. Voloshin, Coloring Mixed Hypergraphs: Theory, Algorithms and Applications, Fields Institute Monographs, AMS 2002

Fact: The corresponding counting functions are polynomials in λ .

Variations on coloring, II

Let $f : V(G) \rightarrow [\lambda]$ be a function, such that Φ is one of the properties below and $\chi_\Phi(G, \lambda)$ denotes the number of such colorings with at most λ colors.

- * **convex:** Every monochromatic set induces a connected graph.
- * **injective:** f is injective on the neighborhood of every vertex.
- **complete:** f is a proper coloring such that every pair of colors occurs along some edge.
- * **harmonious:** f is a proper coloring such that every pair of colors occurs at most once along some edge.
- **equitable:** All color classes have (almost) the same size.
- * **equitable, modified:** All non-empty color classes have the same size.

Fact: For (*), $\chi_\Phi(G, \lambda)$ is a polynomial in λ , for (-), it is not.

Variations on coloring, III

- * **path-rainbow:** Let $f : E \rightarrow [\lambda]$ be an edge-coloring. f is **path-rainbow** if between any two vertices $u, v \in V$ there is a path where all the edges have different colors.

Fact: $\chi_{rainbow}(G, \lambda)$, the number of path-rainbow colorings of G with λ colors, is a polynomial in λ

Rainbow colorings of various kinds arise in computational biology

- * **-monochromatic components:** Let $f : V \rightarrow [\lambda]$ be a vertex-coloring and $t \in \mathbb{N}$. f is an mcc_t -coloring of G with λ colors, if all the connected components of a monochromatic set have size at most t .

Fact: For fixed $t \geq 1$ the function $\chi_{mcc_t}(G, \lambda)$, the number of mcc_t -colorings of G with λ colors, is a polynomial in λ . but not in t .

mcc_t colorings were first studied in:

N. Alon, G. Ding, B. Oporowski, and D. Vertigan. Partitioning into graphs with only small components. *Journal of Combinatorial Theory, Series B*, 87:231–243, 2003.

Variations on coloring, IV

Let $\mathcal{P}$ be any graphs property and let $n \in \mathbb{N}$.

We can define coloring functions $f : V \rightarrow [\lambda]$ by requiring that the union of any n color classes induces a graph in $\mathcal{P}$.

- For $n = 1$ and $\mathcal{P}$ the **empty graphs** $G = (V, \emptyset)$ we get the **proper colorings**.
- For $n = 1$ and $\mathcal{P}$ the **connected graphs** we get the **convex colorings**.
- For $n = 1$ and $\mathcal{P}$ the graphs which are **disjoint unions of graphs of size at most t** , we get the **mcc_t -colorings**.
- For $n = 2$ and $\mathcal{P}$ the **acyclic graphs** we get the **acyclic colorings**, introduced in: **B. Grunbaum, Acyclic colorings of planar graphs, Israel J. Math. 14 (1973), 390-412**

Theorem: Let $\chi_{\mathcal{P},n}(G, \lambda)$ be the number of colorings of G with λ colors such that the union of any n color classes induces a graph in $\mathcal{P}$.

Then $\chi_{\mathcal{P},n}(G, \lambda)$ is a polynomial in λ .

Variations on colorings, V: coloring relations

Let $G = (V, E)$. Here we look at an example where the coloring is a relation $R \subseteq V \times [k]$ rather than a function $f : V \rightarrow [k]$.

We denote by C_v the set $\{c \in [k] : (v, c) \in R\}$.

Let $a, b \in \mathbb{N}$. An (a, b) -coloring relation with k colors is a relation $R \subseteq V \times [k]$ such that

- For each $v \in V$ there are at most a -many colors $c \in [k]$ such that $(v, c) \in R$.
- If $(u, v) \in E$ then $C_u \neq C_v$ and there are **at most** b -many distinct elements $c_1, \dots, c_b$ in $C_u \cap C_v$.

Exercise:

- Compute the number of (a, b) -coloring relations of the complete graphs K_n for various $a, b, k \in \mathbb{N}$.
- Is the number (a, b) -coloring relations with k colors of a graph G a polynomial in a, b or k ?
- Look at the corresponding definitions with "**at most**" replaced by "**at least**" or "**exactly**".

Variations on colorings, VI: Two kinds of colors.

Let $G = (V, E)$.

Here we look at two disjoint color sets $A = [k_1]$ and $B = [k_1 + k_2] - [k_1]$.

The colors in A are called **proper** colorings.

Our coloring is a function $f : V \rightarrow [k_1 + k_2] = [k]$ such that

- If $(u, v) \in E$ and $f(u) \in A$ and $f(v) \in A$ then $f(u) \neq f(v)$.
- We count the number of colorings with $k = k_1 + k_2$ colors such that k_1 colors are in A , i.e., **proper**.

Theorem 20 (K. Dohmen, A. Pönitz and P. Tittman, 2003)

This gives us a polynomial $P(G, k_1, k)$ in k_1 and k .

Coloring Properties

Why are there many chromatic polynomials?

Back to outline of lecture 2

Colorings

Our framework is as follows:

- Let $\mathfrak{M}$ be a finite relational τ -structure with universe M .
- Let $k \in \mathbb{N}$ and $[k] = \{0, \dots, k-1\}$.
- Let f be an r -ary function $f : M^r \rightarrow [k]$.
- We shall look at families $\mathcal{P}$ consisting of triples of the form $(\mathfrak{M}, f, [k])$.

Coloring properties, I

A class of such triples $\mathcal{P}$ is a **coloring property** if

Extension Property: Let $n \leq k_i, i = 1, 2$ and let $(\mathfrak{M}, f, [k_1])$ and $(\mathfrak{M}, f, [k_2])$ be two colorings of $\mathfrak{M}$, using only colors in $[n]$, i.e., $rg(f) \subseteq [n]$.
Then $(\mathfrak{M}, f, [k_1]) \in \mathcal{P}$ iff $(\mathfrak{M}, f, [k_2]) \in \mathcal{P}$.

Isomorphism Property: $\mathcal{P}$ is closed under isomorphisms of colorings.

The isomorphisms property implies the permutation property:

Permutation Property: Let $f : M^r \rightarrow [k]$ be a fixed coloring.
For π is a permutation of $[k]$, we define the coloring f_π by $f_\pi(\bar{a}) = \pi(f(\bar{a}))$.
Then $(\mathfrak{M}_k, f, [k]) \in \mathcal{P}$ iff $(\mathfrak{M}_k, f_\pi, [k]) \in \mathcal{P}$.

Coloring properties, II

If instead of coloring functions f we allow coloring relations

$$R \subseteq M^R \times [k]$$

we need some additional properties:

- (i) A coloring property $\mathcal{P}$ of triples $(\mathfrak{M}_k, R, [k]) \in \mathcal{P}$ is **bounded**, if for every $\mathfrak{M}$ there is a number N_M such that for all $k \in \mathbb{N}$ the set of colors

$$\{x \in [k] : \exists \bar{y} \in M^r R(\bar{y}, x)\}$$

has size at most N_M .

- (ii) A coloring property is **range bounded**, if its range is bounded in the following sense: There is a number $d \in \mathbb{N}$ such that for every $\mathcal{M}$ and $\bar{y} \in M^r$ the set $\{x \in [k] : R(\bar{y}, x)\}$ has at most d elements.

Clearly, if a coloring property is range bounded, it is also bounded.

Coloring properties, III

We denote by

$$\chi_{\mathcal{P}}(\mathfrak{M}, k)$$

the **number** of generalized $k - \mathcal{P}$ -coloring R on $\mathfrak{M}$.

Tomer Kotek, Johann A. Makowsky, and Boris Zilber

On Counting Generalized Colorings

to appear in:

Model Theoretic Methods of Finite Combinatorics,

M. Grohe and J.A. Makowsky eds.

Contemporary Mathematics Series of the AMS (2011)

Conference version in: Computer Science Logic, CSL'08, vol. 5213, (2008), 339-353

[File:p-coloring.tex](#)

Generalized multi-colorings

To construct also graph polynomials in **several variables**, we extend the definition to deal with **several color-sets**, and also call them **generalized chromatic polynomials**.

We say an $(\alpha + 2)$ -tuple

$$(\mathfrak{M}, R, [k_1], \dots, [k_\alpha],)$$

with

$$R \subset M^m \times [k_1]^{m_1} \times \dots \times [k_\alpha]^{m_\alpha}$$

is a **generalized multi-coloring** of $\mathfrak{M}$ for colors $\bar{k}^\alpha = (k_1, \dots, k_\alpha)$.

The **extension** and **isomorphism** property are adapted appropriately to deal also with **unused** color-sets.

By abuse of notation, $m_i = 0$ is taken to mean the color-set k_i is not used in R .

A theorem with an elementary generic proof

THEOREM:

For every $\mathcal{M}$ the counting function $\chi_{\phi(R)}(\mathcal{M}, k)$ is a polynomial in k of the form

$$\sum_{j=0}^{d \cdot |\mathcal{M}|^m} c_{\phi(R)}(\mathcal{M}, j) \binom{k}{j}$$

where $c_{\phi(R)}(\mathcal{M}, j)$ is the number of generalised $k - \phi$ -colorings R with a fixed set of j colors.

Polynomials in $\mathbb{Z}[k]$ with monomials of the form $\binom{k}{j}$ are sometimes called **Newton polynomials**.

In the light of this theorem we call $\chi_{\phi(R)}(\mathcal{M}, k)$ also a

generalised chromatic polynomial.

Proof

We first observe that any generalised coloring R uses at most

$$N = d \cdot |M|^m$$

of the k colors.

For any $j \leq N$, let $c_{\phi(R)}(\mathcal{M}, j)$ be the number of colorings, with a fixed set of j colors, which are generalised vertex colorings and use all j of the colors.

Next we observe that any permutation of the set of colors used is also a coloring.

Therefore, given k colors, the number of vertex colorings that use exactly j of the k colors is the product of $c_{\phi(R)}(\mathcal{M}, j)$ and the binomial coefficient $\binom{k}{j}$. So

$$\chi_{\phi(R)}(\mathcal{M}, k) = \sum_{j \leq N} c_{\phi(R)}(\mathcal{M}, j) \binom{k}{j}$$

The right side here is a polynomial in k , because each of the binomial coefficients is. We also use that for $k \leq j$ we have $\binom{k}{j} = 0$. Q.E.D.

The Classical Graph Polynomials

[Back to outline of lecture 2](#)

Prominent (classical) graph polynomials

- The **chromatic polynomial** (G. Birkhoff, 1912)
- The **Tutte polynomial** and its colored versions (W.T. Tutte 1954, B. Bollobas and O. Riordan, 1999);
- The **characteristic polynomial** (T.H. Wei 1952, L.M. Lihtenbaum 1956, L. Collatz and U. Sinogowitz 1957)
- The various **matching polynomials** (O.J. Heilmann and E.J. Lieb, 1972)
- Various **clique** and **independent set polynomials** (I. Gutman and F. Harary 1983)
- The **Farrel polynomials** (E.J. Farrell, 1979)
- The **cover polynomials** for digraphs (F.R.K. Chung and R.L. Graham, 1995)
- The **interlace-polynomials** (M. Las Vergnas, 1983, R. Arratia, B. Bollobás and G. Sorkin, 2000)
- The various **knot polynomials** (of signed graphs) (Alexander polynomial, Jones polynomial, HOMFLY-PT polynomial, etc)

Applications of classical graph polynomials

As we said before, there are plenty of applications of the classical graph polynomials in

- Graph theory proper and **knot theory**;
- Chemistry and biology;
- Statistical mechanics (Potts and Ising models)
- **Social networks** and **finance mathematics**;
- Quantum physics and quantum computing

And what about the many other graph polynomials
we just learned to construct?

How are the **classical** graph polynomials compared?

There are various ways of comparing graph polynomials:

- By **distinctive power**:

$P(G; \bar{X}) <_{d.p.} Q(G; \bar{X})$ if for any two graphs G_1, G_2 with $Q(G_1; \bar{X}) = Q(G_2; \bar{X})$ we also have $P(G_1; \bar{X}) = P(G_2; \bar{X})$

- By **coefficient computation**:

$P(G; \bar{X}) <_{coeff} Q(G; \bar{X})$ if there is a function F which computes for every G the coefficients of $P(G; \bar{X})$ from the coefficients of $Q(G; \bar{X})$.

- By **substitution instance**.

$P(G; \bar{X}) <_{subst} Q(G; \bar{X})$ if there is a substitution σ of the variables such that for every G $P(G; \bar{X}) = Q(G; \sigma(\bar{X}))$.

Proposition: $P(G; \bar{X}) <_{d.p.} Q(G; \bar{X})$ iff $P(G; \bar{X}) <_{coeff} Q(G; \bar{X})$

If $P(G; \bar{X}) <_{coeff} Q(G; \bar{X})$ then $P(G; \bar{X}) <_{subst} Q(G; \bar{X})$, but not conversely.

How are the **classical** graph polynomials **uniformly** defined?

Most graph polynomials studied in the literature have
several equivalent definitions:

- (i) By **counting generalized colorings**;
- (ii) By **generating functions**;
- (iii) By **subset expansion** formulas;
- (iv) By **recurrence relations**;
- (v) By **counting (weighted) homomorphisms**.

We shall see that, **by imposing SOL-definability**,
(i)-(iii) give the **same class of graph polynomials**,
whereas (iv) and (v) are **special cases** thereof.

Non-definability via complexity assumptions:

Harmonious colorings

A vertex coloring of a graph G with k colors is **harmonious** if it is **proper** and each pair of colors appears at most once along an edge.

The **harmonious index** of a graph G is the smallest k such that there is a harmonious coloring with k colors.

- **J.E. Hopcroft and M.S. Krishnamoorthy** studied harmonious colorings in 1983.
- **B. Courcelle, JAM and U. Rotics** have shown that graph parameters (polynomials) definable in CMSOL can be **computed in polynomial** time for graphs of **tree-width at most k** .
- **K. Edwards and C. McDiarmid** showed that computing the harmonious index is **NP-hard even on trees**.
- So **assuming $P \neq NP$** , the harmonious index is **not CMSOL-definable**, because trees have tree-width 1.

Non-definability via complexity assumptions: Chromaticity

- B. Courcelle, J.A.M. and U. Rotics proved that graph parameters (polynomials) definable in CMSOL in the language of graphs can be computed in polynomial time for graphs of clique-width at most k .
- The Exponential Time Hypothesis (ETH) says that 3 – SAT cannot be solved in time $2^{o(n)}$. It was first formulated by R. Impagliazzo, R. Paturi and F. Zane in 2001.
- F. Fomin, P. Golovach, D. Lokshtanov and S. Saurabh proved that, assuming that ETH holds, the chromatic number $\chi(G)$ cannot be computed in polynomial time.
- Therefore, assuming ETH, the chromatic number and the chromatic polynomial are not CMSOL-definable.

Back to outline of Lecture 3

[File:parameters.tex](#)

Non-definability via complexity assumptions: Other cases

There are many other non-definability results which can be obtained like this, for example graph parameters derived from

- dominating sets, or
- the size of a maximal cut.

Our goal is to prove non-definability
without complexity theoretic assumptions.

Additive and multiplicative graph parameters

with respect to a binary operation $\square$

Let $G_1 \square G_2$ denote the disjoint union of two graphs.

f is **additive** if $f(G_1 \square G_2) = f(G_1) + f(G_2)$.

f is **multiplicative** if $f(G_1 \square G_2) = f(G_1) \cdot f(G_2)$.

For $\square$ the **disjoint union** we have:

- (i) $|V(G)|, |E(G)|, k(G)$ are not multiplicative, but additive.
- (ii) $k(G)$ and $b(G)$ are additive.
 $b(G)$ is the number of 2-connected components of G .
- (iii) $\chi(G)$ and $\omega(G)$ are neither additive nor multiplicative.
- (iv) The number of perfect matchings $pm(G)$ is multiplicative and so is the generating matching polynomial $\sum_k m_k(G) X^k$.
 Note that $m_k(G)$ is not multiplicative.
- (v) The graph polynomials $\chi(G, \lambda)$ and $T(G, X, Y)$ are multiplicative.

Maximizing and minimizing graph parameters

with respect to $\square$

Let $G_1 \square G_2$ denote the disjoint union of two graphs.

f is **maximizing** if $f(G_1 \square G_2) = \max\{f(G_1), f(G_2)\}$.

f is **minimizing** if $f(G_1 \square G_2) = \min\{f(G_1), f(G_2)\}$.

Again for $\square$ the **disjoint union** we have

- (i) The various chromatic numbers $\chi(G)$, $\chi_e(G)$, $\chi_t(G)$ are maximizing.
- (ii) The size of the maximal clique $\omega(G)$ and the maximal degree $\Delta(G)$ are maximizing.
- (iii) The tree-width $tw(G)$ and the clique-width $cw(G)$ of a graph are maximizing.
- (iv) The minimum degree $\delta(G)$, the girth $g(G)$ are minimizing.

The girth is the minimum length of a cycle in G .

Back to outline of Lecture 3

[File:parameters.tex](#)

General Connection Matrices (aka Hankel Matrices): I

Let $\mathcal{C}$ be a class possibly **labeled graphs, hyper-graphs** or **τ -structures**.

Let $\square$ be a binary operation defined on $\mathcal{C}$.

Let G_i be an enumeration of all (labeled) finite graphs (structures) in $\mathcal{C}$.

Let f be graph parameter.

The (full) connection matrix $M(f, \square)$ is defined by

$$M(f, \square)_{i,j} = f(G_i \square G_j)$$

and is called the **Full Connection Matrix of f for $\square$ on $\mathcal{C}$** ,

or just a **connection matrix**.

We denote by $r(f, \square)$ the **rank** of $M(f, \square)$.

We shall often look at infinite submatrices of $M(f, \square)$.

Computing $r(f, \square)$

Proposition:

- (i) If f is multiplicative, $r(f, \square) = 1$.
- (ii) If f is additive, $r(f, \square) = 2$.
- (iii) If f is maximizing or minimizing, $r(f, \square)$ is infinite.
- (iv) For the average degree $d(G)$ of a graph, $r(d, \sqcup)$ is infinite.

Proof: The first three statements are easy.

For $f = d(G)$ we have

$$M(d, \sqcup) = 2 \frac{|E_1| + |E_2|}{|V_1| + |V_2|}.$$

This contains, for graphs with a fixed number e of edges, the Cauchy matrix $(\frac{2e}{i+j})$, hence $r(d, \sqcup)$ is infinite. $\square$.

Back to outline of Lecture 3

$\mathcal{L}$ -smooth operations.

Let $\mathcal{L}$ be a logic.

We say that two graphs G, H are $(\mathcal{L},)_q$ -equivalent, and write $G \sim_{\mathcal{L}}^q H$, if G and H satisfy the same $\mathcal{L}$ -sentences of quantifier rank q .

We say that $\square$ is $\mathcal{L}$ -smooth, if whenever we have

$$G_i \sim_{\mathcal{L}}^q H_i, i = 0, 1$$

then

$$G_0 \square G_1 \sim_{\mathcal{L}}^q H_0 \square H_1$$

This definition can be adapted to k -ary operations for $k \geq 1$.

Proving that an operation $\square$ is $\mathcal{L}$ -smooth may be difficult.

For FOL this can be achieved using

Ehrenfeucht-Fraïssé games also known as pebble games.

Another way of establishing smoothness is via the Feferman-Vaught theorem.

Examples of $\mathcal{L}$ -smooth operations.

- (i) Quantifier-free **scalar** transductions are both FOL and MSOL-smooth.
- (ii) Quantifier-free **vectorized** transductions are FOL but not MSOL-smooth.
- (iii) The **cartesian product** is FOL-smooth but not MSOL-smooth.

This was shown by A. Mostowski in 1952.

- (iv) The **(rich) disjoint union** is both FOL and MSOL-smooth.

The **rich** disjoint union has two additional unary predicates to distinguish the universes.

For FOL this was shown by E. Beth in 1952.

For MSOL this is due to H. Läuchli, 1966, using **Ehrenfeucht-Fraïssé games**

- (v) **Adding modular counting quantifiers preserves smoothness.**

For CMSOL and the **disjoint union** this is due to B. Courcelle, 1990.

NEW: For *CFOL* and the **product** this is due to T. Kotek and J.A.M., 2012.

The Finite Rank Theorem

for graph parameters

THEOREM (Godlin, Kotek, Makowsky 2008):

Let f be a **numeric parameter** or **polynomial** for τ -structures definable in $\mathcal{L}$ and taking values in an integral domain $\mathcal{R}$.

Let $\square$ be an $\mathcal{L}$ -smooth operation.

Then the connection matrix $M(f, \square)$ has **finite rank** over $\mathcal{R}$.

The **Proof** uses a Feferman-Vaught-type theorem for graph polynomials, due to **B. Courcelle, J.A.M. and U. Rotics, 2000**.

Back to outline of Lecture 3

Applications of the Finite Rank Theorem to graph parameters

Disjoint unions

The following graph parameters or not CMSOL-definable because they are maximizing (minimizing) for the disjoint union.

- $\omega(G)$, the **clique number** and $\alpha(G)$, the **independence number** of G .
- The **chromatic number** $\chi(G)$ and the **chromatic index** $\chi_e(G)$.
- The **degrees** $\delta(G)$ (minimal), $\Delta(G)$ (maximal)

The same holds for the average degree $d(G)$, but here we use the fact that the Cauchy matrix growing rank.

The harmonious chromatic polynomial

Recall: A vertex coloring of a graph G with k colors is **harmonious** if it is **proper** and each pair of colors appears at most once along an edge.

$\chi_{\text{harmonious}}(G; k)$ counts the number of harmonious colorings of G with at most k colors.

The **harmonious index** $\chi_{\text{harmonious}}(G)$ of a graph G is the smallest k such that there is a harmonious coloring with k colors.

Let iP_2 be the graph which consists of i disjoint edges (in the language of **hyper-graphs**).

Proposition:

- (i) $\chi_{\text{harmonious}}(iP_2 \sqcup jP_2, k) = 0$ iff $i + j > \binom{k}{2}$.
- (ii) $\chi_{\text{harmonious}}(iP_2 \sqcup jP_2) = \min_k \{i + j \leq \binom{k}{2}\}$.
- (iii) $\chi_{\text{harmonious}}(G; k)$ is **not CMSOL-definable** in the language of hyper-graphs.
- (iv) $\chi_{\text{harmonious}}(G)$ is **not CMSOL-definable** in the language of hyper-graphs.

Three graph polynomials, I

Rainbow polynomial $\chi_{rainbow}(G, k)$ is the number of **path-rainbow connected k -colorings**, which are functions $c : E(G) \rightarrow [k]$ such that between any two vertices $u, v \in V(G)$ there exists a path where all the edges have different colors.

MCC-polynomial For every fixed $t \in \mathbb{N}$, $\chi_{mcc(t)}(G, k)$ is the number of vertex k -colorings $f : V(G) \rightarrow [k]$ for which every color induces a subgraph with a **connected component of maximal size t** .

Convex coloring polynomial $\chi_{convex}(G, k)$ is the number of **convex colorings**, i.e., vertex k -colorings $f : V(G) \rightarrow [k]$ such that every color induces a connected subgraph of G .

Makowsky and B. Zilber (2005) showed that $\chi_{rainbow}(G, k)$, $\chi_{mcc(t)}(G, k)$, and $\chi_{convex}(G, k)$ are graph polynomials with k as the variable.

Path-rainbow connected colorings were introduced by G. Chartrand et al. in 2008.

Their complexity was studied in S. Chakraborty et. al in 2008. $mcc(t)$ -colorings were first studied by N. Alon et al. in 2003.

Note $\chi_{mcc(1)}(G, k)$ is the chromatic polynomial.

Convex colorings were studied by S. Moran in 2007.

Three graph polynomials, II

Proposition: The following connection matrices have infinite rank:

- (i) $M(\sqcup_1, \chi_{rainbow}(G, k));$
- (ii) $M(\sqcup_1, \chi_{convex}(G, k));$
- (iii) For every $t > 0$ the matrix $M(\bowtie, \chi_{mcc(t)}(G, k));$

Proof:

$\chi_{rainbow}(G, k)$: We use that the 1-sum of paths with one end labeled is again a path with $P_i \sqcup_1 P_j = P_{i+j-1}$ and that $\chi_{rainbow}(P_r, k) = 0$ iff $r > k + 3$.

$\chi_{convex}(G, k)$: We use edgeless graphs and disjoint union $E_i \sqcup E_j = E_{i+j}$ and that $\chi_{convex}(E_r, k) = 0$ iff $r > k$.

$\chi_{mcc(t)}(G, k)$: We use the join and cliques, $K_i \bowtie K_j = K_{i+j}$ and that $\chi_{mcc(t)}(K_r, k) = 0$ iff $r > kt$.

Three graph polynomials, III

Corollary:

- (i) $\chi_{rainbow}(G, k)$ and $\chi_{convex}(G, k)$ are not CMSOL-definable in the language of graphs and hypergraphs.
- (ii) $\chi_{mcc(t)}(G, k)$ (for any fixed $t > 0$) is not CMSOL-definable in the language of graphs.
- (iii) In particular the chromatic polynomial is not CMSOL-definable in the language of graphs.

Note: It is however CMSOL-definable in the language of ordered hypergraphs.

Proof:

- (i) The 1-sum and the disjoint union are CMSOL-sum-like and CMSOL-smooth for hypergraphs.
- (ii) The join is only CMSOL-sum-like and CMSOL-smooth for graphs.

Back to outline of Lecture 3