Logics, a reminder

We define logics.

- Vocabularies: The basic relations
- Structures: Interpretations of vocabularies
- Variables: Indivicual variables, relation variables, function variables
- Atomic formulas
- Boolean closures
- Quantifications

Vocabularies

A vocabulary is a (finite) set of basic symbols.

We deal with (possibly many-sorted) relational vocabularies.
The basic symbols are sorts symbols and relation symbols.

Sort symbols: $U_\alpha : \alpha \in \mathbb{N}$

Relation symbols: $R_{i,\alpha} : i \in \mathrm{Ar}, \alpha \in \mathbb{N}$ where Ar is a set of arities, i.e. of finite sequences of sort symbols.

Constant symbols: $c_{\alpha,\beta}$ for $\alpha, \beta \in \mathbb{N}$, where α indicates the sort number.

In the case of one-sorted vocabularies, the arity is just of the form $\underbrace{\langle U, U, \ldots \ldots, U \rangle}_{n}$ which will denoted by n.

Vocabularies are denoted by greek letters $\tau, \sigma, \tau_i, \sigma_i$ with $i \in \mathbb{N}$.

τ-structures, I

τ-structures are interpretations of vocabularies.

More precisely, a τ-structure is a function assigning subsets of cartesian products of a fixed set A to each symbol.

$$\mathfrak{A} : \tau \to A \cup \bigcup_{n=1}^{\infty} \wp(A^n)$$

with the following restrictions:

- $\mathfrak{A}(U_\alpha) = A_\alpha \subseteq A$
- $\mathfrak{A}(U_\alpha) \cap \mathfrak{A}(U_\beta) = \emptyset$ for $\alpha \neq \beta$
- If $i = \langle U_{\alpha_1}, \ldots, U_{\alpha_k} \rangle$ is the arity of $R_{i,\alpha}$ then $\mathfrak{A}(R_{i,\alpha}) \subseteq A_{\alpha_1}, \times \ldots, \times A_{\alpha_k}$
- $\mathfrak{A}(c_{\alpha,\beta}) \in A_\alpha$.

τ-structures, II: Graphs and hypergraphs

Graphs and digraphs: $\tau_{graph} = \{U_1, R_{2,1}\}$.

The elements of the set $\mathfrak{A}(U_1) = V$ are called vertices. The subset $\mathfrak{A}(R_{2,1}) = E \subseteq V^2$ is called the (directed) edge relation.
If E is symmetric, the τ-structures is an undirected graph, otherwise it is a directed graph (aka digraph).
If $(u, u) \in E$ the veretx u has a loop.

Hypergraphs: $\tau_{hgraph} = \{U_1, U_2, R_{\langle 1,2 \rangle,1}\}$

The elements of the set $\mathfrak{A}(U_1) = V$ are called vertices.
The elements of the set $\mathfrak{A}(U_2) = E$ are called edges.
The subset $\mathfrak{A}(R_{\langle 1,2 \rangle,1}) \subseteq V \times E$ is called the undirected incidence relation.

Directed hypergraphs: $\tau_{hgraph} = \{U_1, U_2, R_{\langle 1,2,1 \rangle,1}\}$

The elements of the set $\mathfrak{A}(U_1) = V$ are called vertices.
The elements of the set $\mathfrak{A}(U_2) = E$ are called edges.
The subset $\mathfrak{A}(R_{\langle 1,2,1 \rangle,1}) \subseteq V \times E \times V$ is called the directed incidence relation.

τ-structures, III: Labeled graphs and words

Vertex labeled Graphs: Graphs with ℓ-many vertex labels, $\ell \in \mathbb{N}$:
$\tau_{lgraph} = \{U_1, R_{2,1}, P_1, \ldots, P_\ell\}$,
like graphs but with unary predicates P_i for vertex labels.

Edge labeled Graphs: Graphs with ℓ-many edge labels, $\ell \in \mathbb{N}$:
$\tau_{lgraph} = \{U_1, R_{2,i}\}$ with $i = 1, \ldots, \ell$,
like graphs but with ℓ-many edge relations for edge labels.

Words in Σ^*: Let Σ be a finite alphabet (set).
$\tau_{word} = \{U_1, R_{2,1}, R_{1,a}\}$, $a \in \Sigma$, where
$\mathfrak{A}(R_{2,1})$ is a linear order, and
$\mathfrak{A}(R_{1,a}) \cap \mathfrak{A}(R_{1,b}) = \emptyset$ for $a, b \in \Sigma, a \neq b$, and $\bigcap_{a \in \Sigma} \mathfrak{A}(R_{1,a}) = \mathfrak{A}(U_1)$.
τ_{word}-structures satsifying these conditions are words in Σ^*.

Empty structures

In logic and universal algebra a τ-structure $\mathfrak{A}$ is non-empty, i.e., for at least on sort symbol $U_\alpha \in \tau$ the set $\mathfrak{A}(U_\alpha) \neq \emptyset$.

We allow empty structures!

The reason for not allowing empty structures is the axiomatization of First Order Logic FOL. The axiom

$$\forall x P(x) \Rightarrow \exists x P(x)$$

only holds in non-empty one-sorted τ-structures.

Making structures one-sorted

We can always make τ-structures into one-sorted τ'-structures:

- We replace the sorts $U_\alpha \in \tau$ by one sort $V \in \tau'$.
- We add for each sort $U_\alpha \in \tau$ a unary relation symbol $P_\alpha \in \tau'$.
- We replace each $R_{(\alpha_1,\ldots,\alpha_m),i} \in \tau$ by $R_{m,i} \in \tau'$. Constant symbols remain the same.

We then make a τ-structure $\mathfrak{A}$ into a τ'-structure $\mathfrak{A}'$ by setting

- $\mathfrak{A}'(V) = \bigcup_{U_\alpha \in \tau} \mathfrak{A}(U_\alpha)$, and
- $\mathfrak{A}'(P_\alpha) = \mathfrak{A}(U_\alpha)$
- $\mathfrak{A}(R_{(\alpha_1,\ldots,\alpha_m),i}) = \mathfrak{A}'(R_{m,i})$

Isomorphisms and homomorphisms of τ-structures

Let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-strucures on sets
$A = \bigcup_{\alpha, U_\alpha \in \tau} \mathfrak{A}(U_\alpha$ and $B = \bigcup_{\alpha, U_\alpha \in \tau} \mathfrak{B}(U_\alpha$ respectively.
Let $f : A \rightarrow B$ a function. f is a τ-homomorphism if

- For all $U_\alpha \in \tau$ we have:
  $a \in \mathfrak{A}(U_\alpha)$ iff $f(a) \in \mathfrak{B}(U_\alpha)$.

- For all $R_{(\alpha_1,\ldots,\alpha_m),i} \in \tau$ we have:
  $(a_1, \ldots, a_m) \in \mathfrak{A}(R_{(\alpha_1,\ldots,\alpha_m),i})$ iff $(f(a_1), \ldots, f(a_m)) \in \mathfrak{B}(R_{(\alpha_1,\ldots,\alpha_m),i})$.

- For all $c_\alpha \in \tau$ we have:
  $f(\mathfrak{A}(c_\alpha)) = \mathfrak{B}(c_\alpha)$.

f is a τ-isomorphism if additionally f is one-one and onto.

$\mathfrak{A}$ and $\mathfrak{B}$ are τ-isomorphic if there is a τ-isomorphism $f : A \rightarrow B$.

τ-substructures

Let $\mathfrak{A}$ and $\mathfrak{B}$ be two τ-strucures on sets
$A = \bigcup_{\alpha, U_\alpha \in \tau} \mathfrak{A}(U_\alpha$ and $B = \bigcup_{\alpha, U_\alpha \in \tau} \mathfrak{B}(U_\alpha$ respectively.

$\mathfrak{A}$ is isomorphic to a substructure of $\mathfrak{B}$ if there is a function $f : A \rightarrow B$ such that:

- f is one-one.
- For all $U_\alpha \in \tau$ we have:
  If $a \in \mathfrak{A}(U_\alpha)$ then $f(a) \in \mathfrak{B}(U_\alpha)$.
- For all $R_{(\alpha_1, \ldots, \alpha_m), i} \in \tau$ we have:
  If $(a_1, \ldots, a_m) \in A^m$ then
  $(a_1, \ldots, a_m) \in \mathfrak{A}(R_{(\alpha_1, \ldots, \alpha_m), i})$ iff $(f(a_1), \ldots, f(a_m)) \in \mathfrak{B}(R_{(\alpha_1, \ldots, \alpha_m), i})$.
- For all $c_\alpha \in \tau$ we have:
  $f(\mathfrak{A}(c_\alpha)) = \mathfrak{B}(c_\alpha)$.

If f is the identity, we say $\mathfrak{A}$ is a substructure of $\mathfrak{B}$.

Subgraphs and induced subgraphs

In graph theory an undirected graph G without multiple edges is given by two sets $V(G)$ and $E(G)$ with $E(G) \subseteq V(G)^{(2)}$.

Let G, H be two graphs.

Subgraph: H is a subgraph of G if $V(H) \subseteq V(G)$ and $E(H) \subseteq V(H)^2 \cap E(G)$.

This corresponds to the notion of substructure for graphs viewed as hypergraphs. i.e., τ-structures for $\tau = \tau_{hgraph}$

Induced subgraph: H is an induced subgraph of G if $V(H) \subseteq V(G)$ and $E(H) = V(H)^{(2)} \cap E(G)$.

This corresponds to the notion of substructure for graphs viewed as graphs, i.e., τ-structures for $\tau = \tau_{graph}$

Isomorphisms: H and G are isomorphic as τ_{graph}-structures iff they are isomorphic as τ_{hgraph}-structures.

Properties of a τ-structure

A property of τ-structures is a class $\mathcal{P}$ of τ-structures closed under τ-isomorphisms.

Examples:

- All *finite* τ-structures.
- All $\{R_{2,0}\}$-structures where $R_{2,0}$ is interpreted as a linear order.
- Al finite 3-dimensional matchings $3DM$, i.e. all $\{R_{3,0}\}$-structures with universe A where the interpretation of $R_{3,0}$ contains a subset $M \subseteq A^3$ such that no two triples of M agree in any coordinate.
- All binary words which are palindroms.

We say a τ-structure $\mathcal{A}$ has property $\mathcal{P}$ iff $\mathcal{A} \in \mathcal{P}$.

First Order Logic FOL

We now assume our vocabularies are one-sorted with sort symbol V.

We define the set of formulas FOL(τ):

Variables: $u, v, w, \ldots$ ranging over elements of the interpretation of V.

Terms: Variables and constant symbols in τ are τ-terms.

Atomic formulas: For each $R_{m,j} \in \tau$ and τ-terms $t_1, \ldots, t_m$ the expressions $R_{m,j}(t_1, \ldots, t_m)$, $t_1 = t_2$ are atomic formulas in FOL(τ).

Boolean conncectives: If ϕ and ψ are in FOL(τ), so are $\phi \wedge \psi$, $\phi \vee \psi$, $\phi \Rightarrow \psi$ and $\neg \phi$.

Quantifiers: If ϕ is in FOL(τ) and v is a variable, then $\exists v \phi$ and $\forall v \phi$ are in FOL(τ).

Second Order Logic SOL

We now define SOL(τ), the set of SOL-formulas for a vocabulary τ:

FOL : FOL(τ) $\subseteq$ SOL(τ) and SOL(τ) is closed under boolean connectives and first order quantification.

Second order variables: For each $m, j \in \mathbb{N} - \{0\}$ we have second order variables $X_{m,j}$ of arity m.
For each $X_{m,j}$ a second order variable, and τ-terms $t_1, \ldots, t_m$ the expression $X_{m,j}(t_1, \ldots, t_m)$, is an atomic formulas in SOL(τ).

Second order quantification: If $\phi \in$ SOL(τ) so are $\forall X_{m,j}\phi$ and $\exists X_{m,j}\phi$.

Monadic Second Order formulas MSOL(τ) are those where for the arity m of the second order variables we have $m = 1$.

Analogously, SOL^n(τ) is obtained by restricting the arity m of the second order variables to $m \leq n$.

Outline of the course

LECTURE 00: Second Order Logic (SOL) and its fragments (Background, not lectured) LOGICS (14 slides)

LECTURE 01: Classical graph properties and graph parameters and their definability in SOL (4 hours) G-PARAMETERS, (60 slides)

LECTURE 02: One, two, many graph polynomials (4 hours) LANDSCAPE, (ca. 50 slides)

LECTURE 03: The characteristic and the matching polynomial (4 hours MATCHING, (54 slides)

LECTURE XX: Graph polynomials in Physics and Chemistry (2 hours) CHEMISTRY, (38 slides)