Mathematical Logics
First Order Logic*

Fausto Giunchiglia and Mattia Fumagalli

University of Trento

*Originally by Luciano Serafini and Chiara Ghidini
Modified by Fausto Giunchiglia and Mattia Fumagalli

1 Lecture index

1. Intuition
2. Language
3. Interpretation function
4. Satisfiability with respect to an assignment
5. Satisfiability, Validity, Unsatisfiability, Logical Consequence and Logical Equivalence
6. Exercizes
7. Finite domains
8. Analogy with Data bases

2 Query answering in FOL

- We are interested in knowing the set of objects which share a given property.
- More in general are interested in knowing the set of *n*-tuples of objects which are in a certain *n*-ary relation.
- A property in FOL can be expressed by a formula with free variables $x_1, \ldots, x_n$, in formulas, $\varphi(x_1, \ldots, x_n)$.
- *Examples of queries:*
    - $person(x) \wedge earn(x, y) \wedge y > 1000$: the persons (free variable x) who earns more than 1000 euros
    - $\exists z\, (worksFor(x, z) \wedge worksFor(y, z))$: the pairs of people (the free variables (x, y)) who works in the same project.

3 Query answering in FOL

Query answering

Given an interpretation I (a database instance) of a FOL Language and a formula $\varphi(x_1, \ldots, x_n)$ with n-free variables, find all the n-tuples of elements of the domain $(d_1, \ldots, d_n) \in (\Delta^I)^n$ such that

$$I \models \varphi[a[x_1/d_1 \ldots x_n/d_n]]$$

Note (analogy with relational DBs):

1. $\varphi(x_1, \ldots, x_n)$ represents one DB relation
2. A Relational DB as a set of formulas

Example of interpretation (as from above)

Example (Of interpretation)

Symbols

Constants: *alice*, *bob*, *carol*, *robert*
Function: *mother-of* (with arity equal to 1)
Predicate: *friends* (with arity equal to 2)

Domain

$\Delta = \{1, 2, 3, 4, \dots\}$

Interpretation

$I(alice) = 1$, $I(bob) = 2$, $I(carol) = 3$, $I(robert) = 2$

$$I(\textit{mother-of}) = M \qquad \begin{aligned} M(1) &= 3 \\ M(2) &= 1 \\ M(3) &= 4 \\ M(n) &= n + 1 \text{ for } n \geq 4 \end{aligned}$$

$$I(friends) = F = \left\{ \begin{array}{ccc} \langle 1,2\rangle, & \langle 2,1\rangle, & \langle 3,4\rangle, \\ \langle 4,3\rangle, & \langle 4,2\rangle, & \langle 2,4\rangle, \\ \langle 4,1\rangle, & \langle 1,4\rangle, & \langle 4,4\rangle \end{array} \right\}$$

Example

What is the result of the following queries against the interpretation below?

1. *friends*(*x*, *alice*) $\{2, 4\}$
2. $\neg$*friends*(*x*, *bob*) $\{2, 3, 5, 6\}$
3. *friends*(*x*, *y*) $\wedge$ *friends*(*y*, *z*) $\left\{ \begin{array}{l} (1,2,1), (1,2,4), (2,1,2), (2,1,4), \\ (3,4,3), (4,3,4), (4,2,4), (4,1,4), \\ (4,4,1), (4,4,2), (4,4,3), (4,4,4) \end{array} \right\}$
4. $\exists y$ (*friends*(*x*, *y*) $\wedge$ *friends*(*y*, *z*)) $\left\{ \begin{array}{l} (1,1), (1,4), (2,2), (2,4), \\ (3,3), (4,4), (4,1), (4,2), \\ (4,3) \end{array} \right\}$

The interpretation I is defined as follows:

Symbols

Constants: *alice*, *bob*, *carol*, *robert*
Function: *supervisor* (with arity equal to 1)
Predicate: *friends* (with arity equal to 2)

Domain $\Delta^I = \{1, 2, 3, 4, 5, 6\}$

Interpretation $I(alice) = 1$, $I(bob) = 2$, $I(carol) = 3$, $I(robert) = 2$

$I(supervisor) = S$

$S(1) = 3 \quad S(2) = 1$
$S(3) = 4 \quad S(4) = 5$
$S(5) = 5 \quad S(6) = 5$

$$I(friends) = F = \left\{ \begin{array}{lll} (1,2), & (2,1), & (3,4), \\ (4,3), & (4,2), & (2,4), \\ (4,1), & (1,4), & (4,4) \end{array} \right\}$$

Analogy with Databases

When

- the domain of interpretation Δ is ***finite***,
- … and L does contains ***no functional symbols*** (relational language)
- … and there is the ***UNA***,
- … and facts that are not known (not stated in tables) are assumed to be false (**CWA: Closed world assumption**)

then FOL can be used to formalize relational databases

We have the following correspondences:

- Non logical simbols of L correspond to database schema (tables)
- Δ corresponds to the set of values which appears in the tables (elements of the language and of the domain have the same name)
- the interpretation I of a relation corresponds to the tuples that belong to each relation
- «Certain» formulas on L corresponds to queries over the database
- Interpretation of formulas of L which are queries correspond to answers.

Analogy with Databases

FOL	DB
friends	`CREATE TABLE FRIENDS (friend1 : INTEGER friend2 : INTEGER)`
$friends(x,y)$	`SELECT * FROM FRIENDS`
$friends(x,x)$	`SELECT friend1 FROM FRIENDS WHERE friend1 = friend2`
$friends(x,y) \wedge x = y$	`SELECT * FROM FRIENDS WHERE friend1 = friend2`
$\exists x.friends(x,y)$	`SELECT friend2 FROM FRIENDS`
$friends(x,y) \wedge friends(y,z)$	`SELECT * FROM FRIENDS as FRIENDS1 FRIENDS as FRIENDS2 WHERE FRIENDS1.friend2 = FRIENDS2.friend1`

NOTE:

1. Intended meaning of columns not properly managed (e.g., existential information, doable but more complex)
1. Better formalization of relational DBS and Knowledge Graphs (KGs) via Description Logics (DL)

Mathematical Logics
First Order Logic*

Fausto Giunchiglia and Mattia Fumagalli

University of Trento

**Originally by Luciano Serafini and Chiara Ghidini*
Modified by Fausto Giunchiglia and Mattia Fumagalli