Logic in Computer Science

Frank Wolter

Meta Information

Slides, exercises, and other relevant information are available at:

http://www.liv.ac.uk/~frank/teaching/comp118/comp118.html

The module has

- 18 lectures.
- 5 tutorials (Thursday groups start in week 3, Monday groups in week 4).
- Participation and reasonable attempts to solve problems before tutorials is worth 6 percent of final mark).
- Two class tests (25 minutes each, worth 14 percent of final mark).
- Exam (90 minutes, worth 80 percent of final mark).

If you need to see me individually I will be available in my office (room 114, Ashton Building) on Mondays 4–6pm.

Aims of the Module

- To introduce the notation and concepts of formal logic.
- To describe and emphasise the role of formal logic in Computer Science and Information Systems.
- To promote the importance of formal notations as the necessary means of ensuring clarity, precision and absence of ambiguity.
- To provide a solid foundation for modules that make use of formal logic such as data bases, artificial intelligence, formal methods, knowledge representation, multi-agent systems, ontology languages, and advanced web technologies.

Learning Outcomes

At the end of the module the student should be able to:

- translate natural language descriptions and reasoning processes to and from logical equivalents in the propositional and predicate logic.
- evaluate first-order predicate logic formulae in relational stuctures and understand the relationship to relational databases.
- state and apply a proof system (either tableaux or sequent) for propositional and predicate logic.

The Unusual Effectiveness of Logic in Computer Science

Title refers to a symposium and article (by the same title) held at the 1999 Meeting of the American Association for the Advancement of Science. The paper is co-authored by J. Halpern, R. Harper, N. Immerman. P. Kolaitis, M. Vardi, and V. Vianu. It refers to

- the article *On the unreasonable effectiveness of mathematics in the natural sciences*, by E.P. Wigner (1960), a joint winner of the 1963 Nobel Prize for Physics. `http://www.dartmouth.edu/~matc/MathDrama/reading/Wigner.html`.

Very Brief History of Logic: until 19th century

Until the 19th century, logic dealt with arguments in natural languages. For example, **syllogisms**:

- All humans are mortal. All Greeks are humans. Therefore, all Greeks are mortal.

The conclusion can be drawn without understanding the meaning of the words "human", "mortal", "Greeks". Only the form of the sentences is relevant:

- All B are C. All A are B. Therefore, all A are C.

Another example are **self referential paradoxes**:

- Consider the statement: "This sentence is false."

Is the statement true or false?

Very Brief History of Logic: mid 19th century

George Boole (1847): "The Mathematical Analysis of Logic" attempts to formalise logic in the same way as mathematics formalises the manipulation of equations (and other expressions) with numbers.

For example, the distributive law for numbers:

$$(x + y) \times z = (x \times z) + (y \times z)$$

corresponds to a distributive law in propositional logic (or boolean algebra):

$$(p \vee q) \wedge r \equiv (p \wedge r) \vee (q \wedge r)$$

George Boolos: "The design of the following treatise is to investigate the fundamental laws of the operations of mind by which reasoning is performed."

Logic and Foundations of Mathematics: from late 19th century

In mid 19th century, Mathematics (Geometry, Calculus) had rather shaky foundations. For example, no clear answer could be given to:

- What does it mean that $\sum_{i=1}^{\infty} a_i$ exists?
- Is Euclidean geometry the only possible geometry?

In "Begriffsschrift" (1879), Frege proposed logic as a foundation for mathematics. He invented, among other things, the basics for first-order predicate logic:

- **Constants** such as π;
- **Predicates** such as $<$ to assert $4 < 8$;
- **Functions** such as $+$ to form $1 + 1$;
- **Logical connectives** such as $\wedge$ from Boole;
- **Quantifiers** such as "for all": $\forall$.

Logic and Foundations of Mathematics: from late 19th century

In 1902, Russell found that Frege's systems is **inconsistent**:

In Frege's system, one can form for any property P, the set

$$\{x \mid P(x)\}$$

Consider the property $P = (x \notin x)$.

Then $\{x \mid x \notin x\}$ is a set.

Do we have:

$$\{x \mid x \notin x\} \in \{x \mid x \notin x\}?$$

Note similarity to the self-referential paradox "This sentence is false" from above!

Logic and Foundations of Mathematics: from late 19th century

In **Principia Mathematica** (1913, more than 2000 pages), Russell and Whitehead attempt to repair Frege's system and develop logic as a foundation for mathematics.

The following problems where then formulated by Hilbert:

- Can we **prove** that Principia Mathematica (mathematics) is consistent?
- Can we develop a **complete formal system** for mathematics?
- Is mathematics **decidable** (is there a mechanical way to determine whether a given mathematical statement is true)?

Logic and Foundations of Mathematics: from late 19th century

The answers are **negative** (proofs use again self reference):

- Gödel (1930th): one cannot axiomatize arithmetic;
- Gödel (1930th): one cannot prove the consistency of mathematics;
- Church and Turing (1930th): there is no mechanical procedure that can decide whether a first-order predicate logic sentence is a tautology.

Birth of Computer Science: To prove the results above one has to answer the following questions: what is a mechanical procedure? What is a solvable problem? What is an algorithm?

Try **Logicomix**, a graphic novel dealing with the Foundations of Mathematics and figures such as Russell, Frege, Hilbert, Cantor, Wittgenstein, Goedel, etc.

Logic in Computer Science

Logic occupies a central place in Computer science; it has been called **the calculus of computer science**. The Turing Award (the most prestigious award in computer science) has been awarded for logical methods in computing to

- Hoare (1980): semantics for programming languages;
- Codd (1981): database management systems;
- Cook (1982): complexity of computation;
- Milner (1991): logical formalisms for computing;
- Pnueli (1996): temporal logic in computer science;
- Clarke, Emerson, Sifakis (2007): model checking for verification.

Some applications of logic will be briefly discussed in this module.

Syllabus

- Introduction: the unusual effectiveness of logic in computer science;
- Propositional logic (5 lectures):
  - Reminder: syntax and semantics of propositional logic,
  - SAT, logical consequence, logical equivalence, and normal forms,
  - a proof system for propositional logic.
- Introduction to First-order Predicate Logic (11 lectures):
  - syntax of first-order predicate logic,
  - semantics of first-order predicate logic,
  - evaluating first-order predicate logic and relational databases,
  - a proof system for first-order predicate logic,
  - undecidability of first-order predicate logic.
- Outlook: the unusual effectiveness of logic in computer science (1 lecture)