COMP60121
Automated Reasoning

Renate Schmidt

Andrei Voronkov

School of Computer Science
University of Manchester

`http://www.cs.man.ac.uk/~schmidt/COMP6012/`

Prof. Andrei Voronkov

`http://www.voronkov.com/`

Why Automated Reasoning?

- Artificial Intelligence, Mathematics, Philosophy, Linguistics

  Robbins Algebra Problem, Oct. 1996

  http://www.nytimes.com/

- Software + hardware verification, safety critical applications

  The Pentium Bug

  http://micro.magnet.fsu.edu

  Ariane 5 Failure, 4.6.1996

  http://www.dutchspace.nl/

- Web and agent technologies

Why You May Wish To Take COMP60121

- Inform/support other MSc course units (but not pre/co-requisites):
  - ▸ COMP60161: Knowledge Representation and Reasoning
  - ▸ COMP60462: The Semantic Web: Ontologies and OWL
  - ▸ COMP60391: Computer Security

- MSc in Mathematical Logic and the Theory of Computation

- Core in ACS specialisations:
  - ▸ Formal Methods
  - ▸ Artificial Intelligence

Course Outline

When?

    Period 2, Semester 1

    Thursdays

Where?

    Lectures: 2.19

    Labs: 2.25

A Course of Two Halves:

I Logic, Introduction to AR, Logic Programming (AV)

II Advanced Automated Reasoning (RS)

Part I: Logic, AR and Logic Programming

- Propositional Logic (pre-requisite)

- First-order/Predicate Logic

- Propositional and First-order Reasoning using Resolution
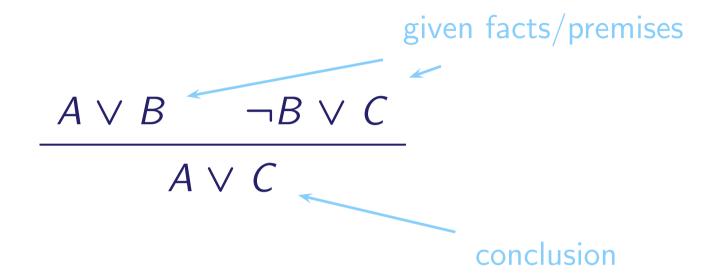
- Logic Programming: Prolog

Reasoning Example

Given facts:

- If I live in Manchester then it is sunny

- If it is sunny then I need a hat

Conclusion:

- If I live in Manchester then I need a hat

The Resolution Principle

given facts/premises

$$\frac{A \vee B \qquad \neg B \vee C}{A \vee C}$$

conclusion

Basis for

- the best Automated Theorem Provers
  e.g. Vampire (Andrei), SPASS

- Logic Programming: Prolog

Logic Programming and Prolog

- Prolog Program — Rules and Facts:

  ```
  has_ancestor(X,Y) :- has_parent(X,Y).
  has_ancestor(X,Y) :-
      has_parent(X,Z), has_ancestor(Z,Y).

  has_parent(roy,sue).
  has_parent(sue,toby).
  ```

- Run program — Query:

  ```
  ?- has_ancestor(roy,X).

  X = sue;
  X = toby;
  ```

Part II: Advanced Techniques

Why?

- The basic resolution calculus is very simple

  - ► Just two rules

  - ► Extremely prolific at generating new conclusions

  - ► Inefficient, impracticable

- Advanced techniques are available

- Part II is devoted to advanced concept of resolution

Advanced Concepts of Resolution

- Avoid unnecessary inferences

- Powerful search control mechanisms

  ▸ Orderings and selection functions

- General notion of redundancy

  ▸ Simplification and optimisation techniques

- Optimised transformations into clausal form

- Has many uses and applications

  ▸ This course: encryption key exchange protocol verfication

Teaching Format

Lectures:

- include Examples Classes

- paper-based exercises and assignments

Labs:

- Approximately 35% of Teaching Time is lab

- Prolog

- try out SPASS

Pre-requisites

- Propositional Logic

- Elementary set theory

  - What is a set, a relation, a function, set operations (intersection, union, etc), properties of binary relations (reflexivity, symmetry, transitivity, etc)

  - Not covered by lectures but will be part of first exercise sheet

  - Exercise sheet available from course website

- Knowledge of first-order logic and some logic programming experience would be some advantage, but is not essential

Reading List

- Recommended elementary level textbook:

  Kelly, J. (1997), *The Essence of Logic*. Prentice Hall.

- Recommended, more advanced:

  Schöning, U. (1989), *Logic for Computer Scientists.* Birkhäuser.

  Fitting, M. (1990), *First-Order Logic and Automated Theorem Proving*. Springer.

- See course unit description for more supplementary texts

Assessment

- Examination (40%)

  - ▸ closed book

- Labs and coursework (30% Part I, 30% Part II)