MSc Module CS612

Automated Reasoning

Who, What, When, Where, Why?

Renate Schmidt
(email: schmidt@cs.man.ac.uk**)**
Alan Williams
(email: alanw@cs.man.ac.uk**)**

September 2005

Why?

- The Pentium Bug

Why?

- The Pentium Bug
- The Pentium II Bug

Why?

- The Pentium Bug
- The Pentium II Bug
- Arriane 5 Failure, 4 June 1996

Why?

- The Pentium Bug
- The Pentium II Bug
- Arriane 5 Failure, 4 June 1996
  - software specification and design errors

Why?

- The Pentium Bug
- The Pentium II Bug
- Arriane 5 Failure, 4 June 1996
  - software specification and design errors
- Internet encryption bug...

Why?

- The Pentium Bug
- The Pentium II Bug
- Arriane 5 Failure, 4 June 1996
  - software specification and design errors
- Internet encryption bug. . . hasn't been found (I made it up!)

Why?

- The Pentium Bug
- The Pentium II Bug
- Arriane 5 Failure, 4 June 1996
  - software specification and design errors
- Internet encryption bug. . . hasn't been found (I made it up!). . . yet

What?

- System description via formal logic

What?

- System description via formal logic
- Analysis and reasoning

What?

- System description via formal logic
- Analysis and reasoning
- Automation

What?

- System description via formal logic
- Analysis and reasoning
- Automation
- Advanced techniques for efficiency; theoretical concepts

Course Outline

When?

Semester 1

Week 7 (w.b. 7th November 2005)

Where?

Lectures: 2.15

Labs: 2.25a

A Course of Two Halves:

1. Formal Logic and Automated Reasoning (AJW)
2. Advanced Automated Reasoning (RenS)

Part of FM Specialisation and AI Specialisation,
No prerequisites

Formal Logic and Automated Reasoning

- Classical Propositional Logic
- First-order Predicate Logic
- Automated Reasoning: Methods and Tools, including
  - resolution
  - logic programming

Assumptions:

Assumptions:

IF I live in Manchester THEN it is raining

Assumptions:

IF I live in Manchester THEN it is raining

IF it is raining THEN I need an umbrella

Assumptions:

IF I live in Manchester THEN it is raining

IF it is raining THEN I need an umbrella

Conclusion:

IF I live in Manchester THEN I need an umbrella

Resolution Principle

Resolution Principle

Assumptions: $(A \vee B)$ $(C \vee \neg B)$

Resolution Principle

Assumptions: $(A \vee B)$ $(C \vee \neg B)$

Conclusion: $(A \vee C)$

Resolution Principle

Assumptions: $(A \vee B)$ $(C \vee \neg B)$

Conclusion: $(A \vee C)$

The basis of

- Automated Theorem-proving
- Logic Programming

Logic Programming and Prolog

```
ancestor(X,Y) :- parent(X,Y).

ancestor(X,Y) :- parent(X,Z),
                 ancestor(Z,Y).

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

Logic Programming and Prolog

```
ancestor(X,Y) :- parent(X,Y).

ancestor(X,Y) :- parent(X,Z),
                 ancestor(Z,Y).

parent(sue,toby).
parent(roy,sue).

?- ancestor(roy,X).

X = sue;
X = toby;
```

Need for Advanced Techniques

- The basic resolution calculus is very simple
  - Just two rules
  - Extremely prolific at generating new conclusions
  - Inefficient, impracticable

Need for Advanced Techniques

- 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 Automated Reasoning

Emphasis in Part II

- Foundations of advanced automated theorem proving
  - Selection of important topics
  - Many examples and exercises

Emphasis in Part II

- Foundations of advanced automated theorem proving
  - Selection of important topics
  - Many examples and exercises
- Two styles of inference systems
  - Resolution: local, "forward"
  - Semantic tableau: global, goal-oriented, "backward"

Emphasis in Part II

- Foundations of advanced automated theorem proving
  - Selection of important topics
  - Many examples and exercises
- Two styles of inference systems
  - Resolution: local, "forward"
  - Semantic tableau: global, goal-oriented, "backward"
- Important basic properties
  - Soundness $\rightsquigarrow$ no false conclusions are drawn
  - Completeness $\rightsquigarrow$ all true conclusions are drawn
  - Efficiency $\rightsquigarrow$ avoid unnecessary inferences

Modern Resolution Framework

- Best provers use resolution

Modern Resolution Framework

- Best provers use resolution
- Modern resolution framework = an extension of basic resolution calculus with:
  - Powerful search control mechanisms
    - $\rightsquigarrow$ ordering and selection refinements
  - General notion of redundancy
    - $\rightsquigarrow$ simplification and optimisation techniques
  - optimised transformations into clausal form

Modern Resolution Framework

- Best provers use resolution
- Modern resolution framework = an extension of basic resolution calculus with:
  - Powerful search control mechanisms
    - ⇝ ordering and selection refinements
  - General notion of redundancy
    - ⇝ simplification and optimisation techniques
  - optimised transformations into clausal form
- Has many uses and applications
  - This course: verfication of Neuman-Stubblebine key exchange protocol
- Fast implementations: Vampire, (M)SPASS

Semantic tableau

- Given by a set of inference rules, e.g.:

$$\frac{F \wedge G}{\begin{array}{l} F \\ G \end{array}} \qquad \frac{F \vee G}{F \quad | \quad G}$$

- Used to construct derivation trees
- Basis for semantic tableau provers

Content

- More details are in the Syllabus

Topics of Current Research

- Developing practical decision procedures
- Handling specific theories (equality, transitive relations, . . .) or logics (description logics, modal logics, . . .)
- Implementing fast automated theorem provers
- Relationship between different proof methods (resolution & tableau, . . .)
- Combining different proof methods and different provers
- Specific applications:
  - Software engineering
  - Ontologies and the semantic web
  - Multi-agent systems

Labs:

- Approximately 40% of teaching week is lab
- Prolog
- MSPASS, Vampire

Reading List

- Needed for pre-coursework:

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

- Recommended:

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

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

Assessment

- Examination (40%)
  - open book
- Teaching Week exercises (30%)
- Post-course assignment (30%):
  - a choice of mini-projects and extended exercises
  - practical and paper-based

Preliminary Work

Introductory Meeting: Monday October 3rd, 12:30–1:00pm (provisional!)

- Course notes
- Background reading
  - Kelly: introduction to reasoning
- Prolog
- Exercises