The Reasoning and Optimization Theme

Joshua Knowles, Konstantin Korovin and Renate Schmidt

School of Computer Science
The University of Manchester

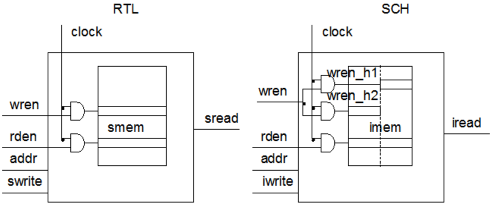
September 16, 2014

Outline

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

Hardware: Are these two hardware designs equivalent?



Software: Does your program accesses unallocated memory?

Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

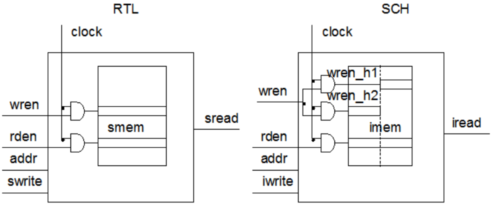
Knowledge management:
Can we represent and analyse all available knowledge about human body ?

Automated reasoning: can we solve all these problems automatically ?

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

  Hardware: Are these two hardware designs equivalent?



  Software: Does your program accesses unallocated memory?

  Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

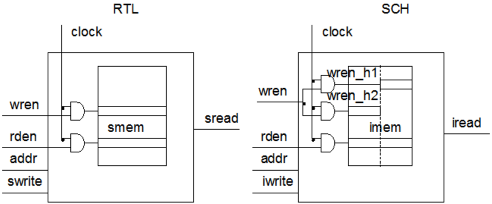
  Knowledge management:
  Can we represent and analyse all available knowledge about human body ?

  Automated reasoning: can we solve all these problems automatically ?

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

  Hardware: Are these two hardware designs equivalent?



Software: Does your program accesses unallocated memory?

Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

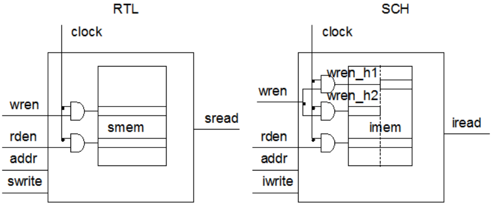
Knowledge management:
Can we represent and analyse all available knowledge about human body ?

Automated reasoning: can we solve all these problems automatically ?

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

  Hardware: Are these two hardware designs equivalent?



  Software: Does your program accesses unallocated memory?

  Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

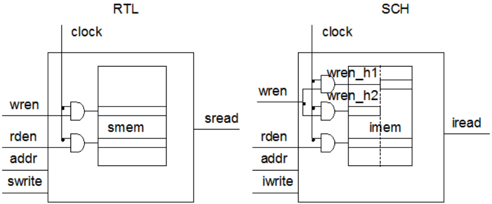
  Knowledge management:
  Can we represent and analyse all available knowledge about human body ?

  Automated reasoning: can we solve all these problems automatically ?

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

  Hardware: Are these two hardware designs equivalent?



  Software: Does your program accesses unallocated memory?

  Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

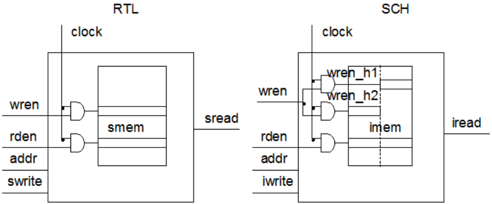
  Knowledge management:
  Can we represent and analyse all available knowledge about human body ?

  Automated reasoning: can we solve all these problems automatically ?

Automated Reasoning

- What is Reasoning? Solving problems by syntactic manipulations.

  Hardware: Are these two hardware designs equivalent?



  Software: Does your program accesses unallocated memory?

  Math: Does this equation $(xy)^{-1} = y^{-1}x^{-1}$ hold in all groups?

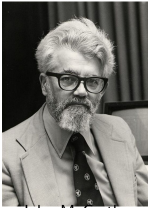
  Knowledge management:
  Can we represent and analyse all available knowledge about human body ?

  Automated reasoning: can we solve all these problems automatically ?

Applications of automated reasoning

Applications:

- software and hardware verification: Intel, Microsoft
- information management: biomedical ontologies, semantic Web, databases
- combinatorial reasoning: constraint satisfaction, planning, scheduling
- Internet security
- Theorem proving in mathematics



John McCarthy

"It is reasonable to hope that the relationship between computation and mathematical logic will be as fruitful in the next century as that between analysis and physics in the past."
McCarthy, 1963.

Manchester: world leading in logic and reasoning

- Theory:
  - first-order reasoning
  - resolution, superposition, instantiation, tableaux, linear arithmetic
  - ontology reasoning
- Applications:
  - software/hardware verification
  - semantic Web, bio-health
  - multi-agent systems
- Reasoning systems developed in our School:
  - iProver – an instantiation-based reasoner for first-order logic won major of awards at CASC championships.

  - Vampire – a superposition-based reasoner for first-order logic, won major awards at CASC championships.

  - MSPASS – a resolution/superposition based reasoner SPASS extended with reasoning with modal logics.
  - Fact++ an ontology reasoner: OWL DL.
  - Pellet an ontology reasoner: OWL DL.

Manchester: world leading in logic and reasoning

- Theory:
  - first-order reasoning
  - resolution, superposition, instantiation, tableaux, linear arithmetic
  - ontology reasoning
- Applications:
  - software/hardware verification
  - semantic Web, bio-health
  - multi-agent systems
- Reasoning systems developed in our School:
  - iProver – an instantiation-based reasoner for first-order logic won major of awards at CASC championships.

  - Vampire – a superposition-based reasoner for first-order logic, won major awards at CASC championships.

  - MSPASS – a resolution/superposition based reasoner SPASS extended with reasoning with modal logics.
  - Fact++ an ontology reasoner: OWL DL.
  - Pellet an ontology reasoner: OWL DL.

Manchester: world leading in logic and reasoning

- Theory:
  - first-order reasoning
  - resolution, superposition, instantiation, tableaux, linear arithmetic
  - ontology reasoning
- Applications:
  - software/hardware verification
  - semantic Web, bio-health
  - multi-agent systems
- Reasoning systems developed in our School:
  - iProver – an instantiation-based reasoner for first-order logic won major of awards at CASC championships.

  - Vampire – a superposition-based reasoner for first-order logic, won major awards at CASC championships.

  - MSPASS – a resolution/superposition based reasoner SPASS extended with reasoning with modal logics.
  - Fact++ an ontology reasoner: OWL DL.
  - Pellet an ontology reasoner: OWL DL.

COMP60332 – Automated Reasoning and Verification

This course is focused on efficient automated reasoning.

This course is self-contained but assumes that students are comfortable with mathematical notions.

Syllabus:

- Propositional logic: syntax, semantics, CNF transformation
  Resolution
  DPLL algorithm: unit propagation, backjumping, lemma learning
- First-order logic: syntax, semantics, Skolemization
  First-order resolution
  How to prove all mathematical theorems using only two rules?
  Elegant mathematical framework for completeness
  Model construction, well-founded induction
  How to make reasoning efficient: redundancy elimination
  What is inside a theorem prover ?
- Applications: verification

COMP60332 – Automated Reasoning and Verification

This course is focused on efficient automated reasoning.

This course is self-contained but assumes that students are comfortable with mathematical notions.

Syllabus:

- Propositional logic: syntax, semantics, CNF transformation
  Resolution
  DPLL algorithm: unit propagation, backjumping, lemma learning
- First-order logic: syntax, semantics, Skolemization
  First-order resolution
  How to prove all mathematical theorems using only two rules?
  Elegant mathematical framework for completeness
  Model construction, well-founded induction
  How to make reasoning efficient: redundancy elimination
  What is inside a theorem prover ?
- Applications: verification

COMP60332 – Automated Reasoning and Verification

This course is focused on efficient automated reasoning.
This course is self-contained but assumes that students are comfortable with mathematical notions.

Syllabus:

- Propositional logic: syntax, semantics, CNF transformation
  Resolution
  DPLL algorithm: unit propagation, backjumping, lemma learning
- First-order logic: syntax, semantics, Skolemization
  First-order resolution
  How to prove all mathematical theorems using only two rules?
  Elegant mathematical framework for completeness
  Model construction, well-founded induction
  How to make reasoning efficient: redundancy elimination
  What is inside a theorem prover ?
- Applications: verification

COMP60332 – Automated Reasoning and Verification

This course is focused on efficient automated reasoning.

This course is self-contained but assumes that students are comfortable with mathematical notions.

Syllabus:

- Propositional logic: syntax, semantics, CNF transformation
  Resolution
  DPLL algorithm: unit propagation, backjumping, lemma learning
- First-order logic: syntax, semantics, Skolemization
  First-order resolution
  How to prove all mathematical theorems using only two rules?
  Elegant mathematical framework for completeness
  Model construction, well-founded induction
  How to make reasoning efficient: redundancy elimination
  What is inside a theorem prover ?
- Applications: verification

COMP60332 – Automated Reasoning and Verification

This course is focused on efficient automated reasoning.

This course is self-contained but assumes that students are comfortable with mathematical notions.

Syllabus:

- Propositional logic: syntax, semantics, CNF transformation
  Resolution
  DPLL algorithm: unit propagation, backjumping, lemma learning
- First-order logic: syntax, semantics, Skolemization
  First-order resolution
  How to prove all mathematical theorems using only two rules?
  Elegant mathematical framework for completeness
  Model construction, well-founded induction
  How to make reasoning efficient: redundancy elimination
  What is inside a theorem prover ?
- Applications: verification

Assessment

Exam: 50%

Closed book, 2 hours, choose 3 out of 4 questions

Coursework and lab: 50%

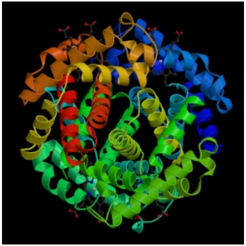
Assessed and unassessed exercises: pen and paper

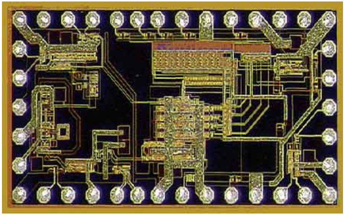
Labwork involving

- SAT solvers
- first-order theorem provers

Questions? please email Konstantin Korovin: korovin@cs.man.ac.uk

Why Optimization?

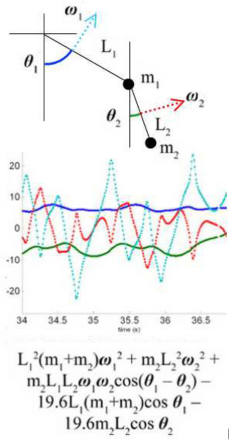




...we recognize efficiency, strength, compactness in these designs. How can we mechanize the design process?

Why Optimization?

- It means finding *the best*; a fundamental aim in all human endeavour
- Optimization saves and makes money; underpins engineering; helps organize, manage and plan; supports machine learning; solves problems
- Computers can do it *fast*
- It is deeply linked with Reasoning by the structure of problems
- It is philosophically linked with mathematics, intelligence, computation, e.g., through the theory of NP-completeness

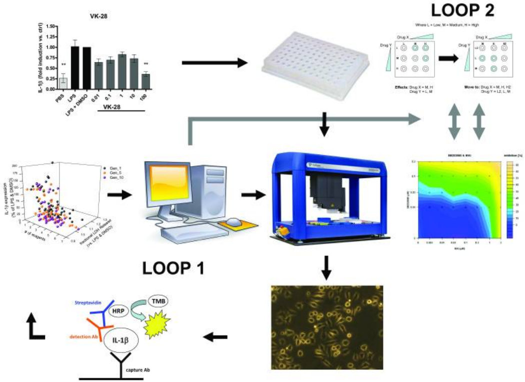


Hamiltonian of a double pendulum discovered by computer

Computer Derives Physical Laws by Itelf

In 2009, Cornell researchers, Schmidt and Lipson, used an *optimization method* based on advanced *symbolic regression* to “discover” physical laws automatically from motion data. (*Science, 2009*)

Optimization in Machine Learning



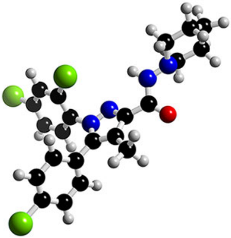
Researchers **at Manchester** in 2011, developed optimization techniques to discover combination therapies to treat brain inflammation, a major factor in Alzheimer's and other diseases. (*Nature Chemical Biology, 2011*)

Optimization in Planning

The Hubble Space Telescope is a scarce resource.

Advanced optimization algorithms are used to schedule time on the telescope, sharing the resource between different groups and taking account of *constraints*

Optimization Saves Money



IcoSystem (a US optimization and complexity science company) set out to improve drug development processes at Eli Lilly.

> 'Compared to a standard 40 months and roughly £25 million, ...[we] reached the same amount of technical success in a year and a day with a total expenditure on the order of £2.7 million.' –Neil Bodick, COO, Eli Lilly Chorus

Optimization in Problem Solving

What is the minimum number of moves (from here) to obtain a solved cube?

What about from ANY configuration?

COMP60342 – What You Will Learn

The course aims to have a *practical* outcome

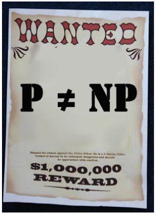
You will learn

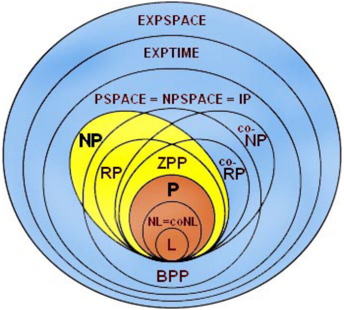
How to approach a wide variety of problems, and how to develop *your own code* to solve them

You will understand some of the most general and flexible optimization algorithms, and how and why they work

What You Will Learn 2

Some important and interesting mathematical theory behind the practice





This links optimization fundamentally to computation and the notion of *complexity*

How You Will Learn It

- Answering thought-provoking questions (often not assessed)
- Doing LAB work, where YOU MUST CODE for yourself
- Running experiments in LABs where you test and compare methods

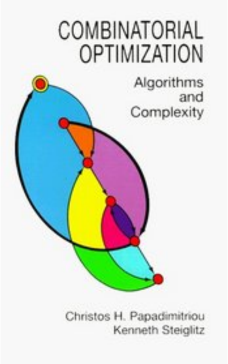
Structure of the Course

Week 1:	Introduction to Optimization
	Problems: from MAX-SAT to TSP to Optimal Betting
	Basic exact methods: Enumeration and Greedy
Week 2:	Branch-and-Bound
	Complexity/NP-Completeness
Week 3:	Dynamic Programming
	Stochastic DP Problems
Week 4:	Evolutionary Algorithms and Local Search
Week 5:	Multiobjective Optimization

Assessments:

1st Lab (two weeks)	Packing a Knapsack with Applications in Finance	10%
2nd Lab (one week)	Hospital Resource Management, and Optimal Stopping	10%
3rd Lab (one week)	Assigning Students to Projects (Project 'Dating')	10%
4th Lab (one week)	Multiobjective Maximum Satisfiability	10%
Examination	Multiple Choice plus 2 Questions from 5 (2 hours)	60%

The Course Text



Costs about £10. Copies available in School and University Library.

Theme outline

Semester 2

Period	Course units
P3	COMP60332 – Automated Reasoning and Verification Konstantin Korovin, Renate Schmidt
P4	COMP60342 – Optimization for learning, planning and problem-solving Joshua Knowles

Teaching day: Friday
Lectures: 2.15

Some advice on choosing themes

The R&O theme can be combined with any other theme

Has no prerequisites, no pre/co-requisite to any theme

It goes well with these themes

- Advanced Web Technologies
- Data Engineering
- Managing Data
- Learning from Data
- Security
- Software Engineering

Core in the Semantic Technologies Pathway

Finally. . .

- Contacting us

  If you are unsure about your theme selection do contact us, either in our offices or send us an email

Joshua Knowles	Room G13, j.knowles@manchester.ac.uk
Konstantin Korovin	Room 2.40, korovin@cs.man.ac.uk
Renate Schmidt	Room 2.42, schmidt@cs.man.ac.uk

- Up-to-date details

  See timetable and course unit descriptions on the web

  See also separate webpages for individual course units: contain additional material such as slides, links to tools, etc