COMP30026 Models of Computation

Review Lecture

Harald Søndergaard

Lecture 22

Semester 2, 2017

Propositional Logic

Propositional formulas: Syntax and semantics.

Semantics is simple, in principle—just a matter of constructing truth tables.

However, it is useful to develop an understanding of the algebraic rules, how to rewrite formulas to equivalent formulas in normal form, and so on.

Important concepts: Satisfiability and validity of formulas, logical consequence and equivalence.

Normal forms: CNF and DNF.

Mechanical proof: Propositional resolution.

First-Order Predicate Logic

Syntax and semantics.

Important semantic tools: the concepts of interpretation, and of an interpretation satisfying a formula (making it true).

Components of an interpretation: Domain and mappings giving meaning to relation symbols, function symbols, constants.

To define the meaning of quantifiers we also need to consider valuations.

Important concepts: Models and counter-models, satisfiability and validity, logical consequence and equivalence.

First-Order Predicate Logic

Useful to develop an understanding of how formulas can be rewritten, rules of passage for the quantifiers and so on.

Normal forms: Clausal form.

Obtaining equi-satisfiable formulas in clausal form: Skolemization.

Mechanical proof: Resolution, including unification.

Proof

There is an expectation that you can provide readable and valid proofs for simple assertions (about the material covered in the subject).

The proofs will not call for induction, even though we have discussed important induction techniques in the subject: Mathematical and structural induction, including more general forms of mathematical induction.

Discrete Mathematics: Sets and Relations

Set operations, algebra of sets.

Binary relations, domains, ranges.

Properties of relations, including
reflexivity, symmetry, anti-symmetry, transitivity.

Total and partial orders.

Equivalence relations.

Discrete Mathematics: Functions

Domain, co-domain, and range of a function.

Image of a set under a function.

Properties of functions, including
injectivity, surjectivity, bijectivity.

Inverse functions.

Regular Languages

Finite-state automata: DFAs and NFAs.

Finite-state automata as recognisers.

The regular operations.

Regular expressions.

Closure properties of regular languages.

Important techniques: Translating NFAs to DFAs, regular expressions to NFAs, and vice versa.

Using the pumping lemma for regular languages to prove non-regularity.

Context-Free Languages

Context-free grammars, derivations of sentences.

Parse trees, ambiguity.

Push-down automata.

(Lack of) closure properties of context-free languages.

Important techniques: Translating a CFG to an equivalent PDA.

Using the pumping lemma for context-free languages to prove languages non-context-free.

Computability, Turing Machines

We based our concept of "computable" on the Turing machine model.

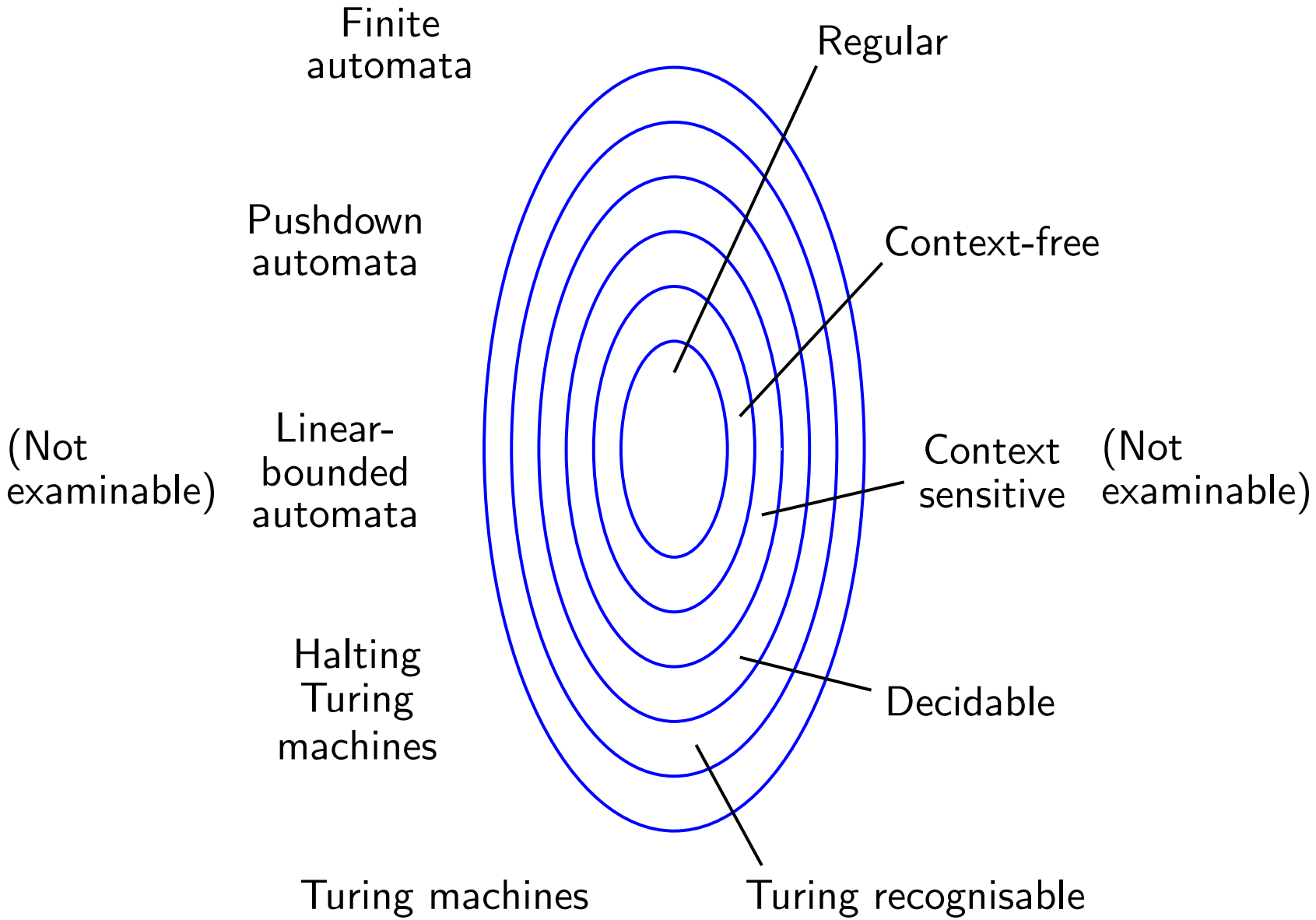
We could have used any other of a large number of equivalent models (partial recursive functions, Markov algorithms, register machines, ...)

The Church-Turing thesis:
Computable is what a Turing machine can compute.

Decidable languages: Those that are recognised by a Turing machines which halts for all input.

Turing recognisable languages: Those that have a Turing machine that acts as a recogniser (and does not necessarily halt).

Models of Computation



Closure Properties

	$\cup$	$\circ$	*	$R\cap$	$\cap$	compl
Regular	Y	Y	Y	Y	Y	Y
Det. context-free	N	N	N	Y	N	Y
Context-free	Y	Y	Y	Y	N	N
Decidable	Y	Y	Y	Y	Y	Y
Turing recognisable	Y	Y	Y	Y	Y	N

Here '$\circ$' means concatenation, '*' means Kleene star, and '$R\cap$' means "intersection with a regular language".

'Det. context-free' means the class of languages that can be recognised by deterministic PDAs (DCFLs).

Decidability of Language Properties

Question	Reg	DCFL	CFL	Decidable	Recognisable
$w \in L$	D	D	D	D	U
$L = \emptyset$	D	D	D	U	U
$L = \Sigma^*$	D	D	U	U	U
$L_1 = L_2$	D	D	U	U	U
$L_1 \subseteq L_2$	D	U	U	U	U
$L =$ given R	D	D	U	U	U
L regular	D	D	U	U	U

Here 'D' = decidable; 'U' = undecidable.

The case of DCFL equality was settled relatively late.

Proof Techniques for (Un-)Decidability

Diagonalisation.

Reduction.

Simulation.

Exploitation of closure properties.

Haskell

Basic Haskell programming is examinable.

It is possible that an exam question will ask you to read or write simple Haskell code, for example to implement some discrete maths concept.

Minor syntax errors will be ignored.

Unless otherwise specified, you can assume that you are free to use functions from the Haskell Prelude, but not from other libraries.

Arrangements Up Until the Exam

Don't forget to take the Student Experience Survey.

I have booked the H. Wilson Theatre on Thursday 26 October from 13:15 for a catch-up session. I imagine we will use it for 90 minutes or so.

The LMS Discussion Forum will stay open until the exam.

Preparing for the Exam

A list of examinable material is on the COMP30026 LMS site.

After this week, practice! There is a practice exam paper, but also go over the tute questions again, and possibly old exam papers (see the "Exam Information" section on the LMS).

Make sure you know where and when the exam is held.

Get a good night's sleep before the exam.

On the Day of the Exam

The exam is a 3-hour closed book exam; reading time is 15 minutes.

Make sure you bring your student card and writing tools.

Be aware that security is now very strict; make sure you comply with all the rules.

The Exam Paper

The format is similar to the practice paper's, and to the mid-semester test's.

You write answers in designated spaces in the exam paper itself. University assessment rules stipulate that you should use blue or black ink.

(You can use a pencil for rough work on the reverse pages.)

Read the COMP30026 LMS site's 'Exam Information' section for more detail. It includes the front page of the exam paper, with detailed instructions for students.

Arrangements after the Exam

Between the exam and the release of results: Examiners are not allowed to discuss any aspect of the exam.

Also, we are not allowed to provide results via the phone or email.

The Last COMP30026 Slide!

Questions?