Lecture 29: Review

CSCI 101
Spring, 2019

Kim Bruce

Decision Problems

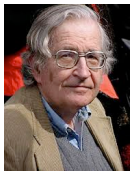
- Are these in D, SD/D, or not-SD
  - 21.1c: {<M> | L(M) = {a}}
  - 21.1p: {<M> : there exists a string w such that |w| < |<M>| and that M accepts w}
  - 21.1q: {<M> : M does not accept any string that ends with 0}
  -

History

Finite Automata

- McCulloch and Pitts (1943) modeling nerves
- Kleene: regular sets, 1956
- Rabin & Scott: non-determinism, 1959
  - Awarded Turing award for paper
  - both students of Church

Context-free Languages



- Chomsky 1956: Modeling natural languages

- Backus-Naur Form: Describing Algol 60
- Oettinger 1961, Schuzenberger 1963: PDA's

Applications of CFL's

- Describing syntax of PL's
  - Can build parser (pda) directly from grammar
- Restrict to deterministic CFL's
  - LL(k), LR(k) limit amount of lookahead to k characters to decide which rule to apply.
- Compiler technology shows great success of theory!!
  - Formal specs of syntax and type-rules can be converted automatically to algorithms!

Context-Sensitive Languages

- Like type 0, but require if $\alpha \rightarrow \beta$ is a rule then $|\beta| \geq |\alpha|$
- Example:
  - $S \rightarrow aSBC$
  - $S \rightarrow aBC$
  - $CB \rightarrow BC$
  - $aB \rightarrow ab$
  - $bB \rightarrow bb$
  - $bC \rightarrow bc$
  - $cC \rightarrow cc$
- Generates $a^n b^n c^n$

Context-Sensitive Languages

- Alternatively, restrict to productions of form
  - $\alpha A \beta \rightarrow \alpha \gamma \beta$
  - Rewrite non-terminals in particular contexts
- Accepted by linear-bounded automata
  - Limit tape to amount of space to write input.
- Interesting from theoretical point of view, but not many applications

Logic of Programs

- Floyd 1967 "Assigning meaning to programs"
  - Flow charts. Turing award 1978 (parsing, semantics, verification, algorithms)
- Hoare 1969 "An axiomatic basis for computer programming"
  - Hoare triples. Turing award 1980 (def & design programming languages)

Logic of Programs

- Remains hard!
  - Especially unsuccessful if write program first and proof later.
  - Systems (Coq) where write proof and extract program from proof.

Model Checking

- In early 1980's: Clarke and Emerson
- Remarkably successful, esp. w/hardware
- Take finite state machine and find which nodes satisfy various assertions.
  - Languages include temporal and fixed-point logics
  - State-space explosion still problem
  - Also useful in concurrency: safety and liveness

Final Exam

- Focus on material not on midterm:
  - Parsing, interpreters, type-checking
  - Lambda calculus, μ-recursive functions, etc.
  - FSM, PDA, TM, Grammars & Languages
  - Decidability & Semidecidability

Final Exam

- 24 hour take-home
- Open book, notes, course web pages, but no discussions w/anyone!
- Start as early as 9 a.m. Monday, must be in no later than 4:30 p.m. next Wednesday.
- 5 questions (multiple parts), emphasis on material not on midterm