Tractable vs. Intractable vs. Undecidable

MIEIC, 2nd Year

João M. P. Cardoso
Email: jmpc@acm.org

U.PORTO
FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO
DEI DEPARTAMENTO DE ENGENHARIA INFORMÁTICA

Tractable *vs*. Intractable Problems

- The problems that cannot be solved with any polynomial time algorithm are called ***intractable* problems**

Undecidable Problems

- A problem is called ***undecidable*** — if no algorithm can produce the correct answer in every instance

Regular Languages (RLs)

Is the language empty?

- L is represented by a *regular expression* or by an *automaton*
- Automata
  - The question resumes to the accessibility in the respective graph: if none final state can be accessed starting in the initial state, the answer is positive
  - Algorithm proportional to the number of edges in the graph: **O(n^2)**
- Regular expression (length n)
  - Convert to ε-NFA, resulting in **O(n)** states, **O(n)** algorithm, or
  - Inspecting regular expression (in the case to include ∅)

Language includes word w?

- L represented by an automata
  - DFA
    - Simulate the processing of the word and accept if processing terminates in a final (acceptance) state and the word was completely processed: **O(|w|)**
  - NFA, ε-NFA
    - Convert to DFA and apply previous method; algorithm maybe exponential in the size of the representation
    - Simpler and more efficient is to simulate the NFA directly, maintaining the set of states in which the automata can be in each transition: **$O(|w|s^2)$**
- L represented by regular expression of size s
  - Convert to ε-NFA with a maximum of *2s* states in **O(s)** and apply the previous method

Context-Free Languages (CFLs)

Tractable

- Test of the empty language:
  - Verify if start variable generates
    - With an adequate data structure: **O(n)**
    - See Hopcroft, Motwani, and Ullman book
- Test if String belongs to a CFL
  - E.g., using CYK algorithm: **$O(n^3)$**

Undecidable Problems

- There are no algorithms for answering to the following questions:
  - Is a given CFG ambiguous?
  - Is a given CFL inherently ambiguous?
  - The intersection of two CFLs is an empty language? (i.e., *L(G*1*)* ∩ *L(G*2*) is* nonempty)
  - Two given CFLs define the same language?
  - A given CFL is the language Σ^*, where Σ is the alphabet? (i.e., is *L(G)* = $\sum^*$?)

Undecidable Problem

- Here is an algorithm that would be easy to implement on a Turing machine:
  - n = 4
  - while (n is the sum of two primes)
    - n = n+2
- "If you could decide whether this program continues forever, and prove your answer, you would be famous! The statement that every even integer greater than 2 is the sum of two primes is known as *Goldbach's conjecture*, named after a mathematician who made a similar conjecture in 1742; mathematicians have been interested in it ever since, but in spite of a $1,000,000 prize offered in 2000 to anyone who could prove or disprove it by 2002, no one has determined whether it is true or false. It is clear, however, that it is true precisely if the program above loops forever."
- "The halting problem is a famous undecidable problem. As the above three-line program might suggest, it is also a good example of a problem for which a decision algorithm, though impossible, would be useful."