Pina Kolling
piko0011

Summary: Lecture 6

Summary for the chapter *8.2.* [1, 3]

Completeness

Let C be a complexity class and let L be a language in C. L is called *C-complete* if any language $L' \in C$ can be reduced to L.

(Every language of a complexity class can be reduced to L.)

- reducitbility is transitive → problems are ordered by difficulty
-

TODO
Questions:
Which problems can be reduced to a formal language?

SAT can be expressed as formal language. [2]

Because CIRQUIT SAT can be reduced to SAT: CIRQUIT SAT can be reduced to a formal language. (?)

Where to set the line between lanuguage decisions and other problems?

What does completeness do for us?

- a reduction definition is usefull because the complexity classe are closed under reduction
- examples look helpfull
- L and R seem to be important:

$$\begin{array}{cc} L' \in P & A \\ L \rightarrow L' & R \end{array}$$

- drawing set circle inclusion thing (P and NP)

TODO
Questions:

P-completeness of CIRCUIT VALUE

Problem: Circuit Value

The CIRCUIT VALUE Problem is the problem of computing the output of a given Boolean circuit on a given input.

In terms of time complexity, it can be solved in linear time (topological sort).

- P-complete
- limit of power of reductions

- got a little tired and zoned out

TODO
Questions:

The reduction (?)

Problem: Circuit Sat

The circuit satisfiability problem (Circuit Sat) is the decision problem of determining whether a given Boolean circuit has an assignment of its inputs that makes the output true.

Input: a Boolean circuit C

Question: Is there a truth assignment which makes C output the value true?

CIRCUIT SAT is NP-complete

- circuit decides nondeterministically (?)
- a variable is added n the nondeterministic Turing Machine
- check if one of the variables is tue: use this choice (?)
- problem: can we set thiese variables such that the Turing Machine accepts?
- answer corresponds direct to *is there a choice of nd decisions such that the turing machine accepts?*
- extremely direct reduction
- cooks theorem :)
- SAT is NP-complete

TODO
Questions:

References

[1] Martin Berglund. *Lecture notes in Computational Complexity.*

[2] klaus-joern Lange. “The Boolean Formula Value Problem as Formal Language”. In: (Jan. 2012). DOI: 10.1007/978-3-642-31644-9_9.

[3] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley Publishing Company, 1994.