Complexity Theory

Georg Moser

Institute of Computer Science @ UIBK

Summer 2008

Outline

- Summary of Last Lecture: Logspace Computability
- Exercises
- Circuit Value Problem
- The Cook-Levin Theorem

Completeness

Definition

hardness

a set $A \subseteq \Sigma^*$ is $\leqslant_m^{\log}$-hard for a complexity class $\mathcal{C}$ if

- $\forall B \in \mathcal{C}$ we have $B \leqslant_m^{\log} A$

Definition

completeness

a set $A \subseteq \Sigma^*$ is complete for $\mathcal{C}$ with respect to $\leqslant_m^{\log}$ if

1. A is $\leqslant_m^{\log}$-hard for $\mathcal{C}$
2. $A \in \mathcal{C}$

Definition

MAZE

- given a directed graph $G = (V, E)$ and nodes $s, t \in V$
- determine whether there is a directed path from s to t in G
- this problem is called MAZE or directed graph reachability

Theorem

Jones, Lien, and Lasser (1976)

MAZE is $\leqslant_m^{\log}$-complete for NLOGSPACE

Corollary

MAZE $\in$ LOGSPACE if and only if LOGSPACE $=$ NLOGSPACE

Homework

1. Homework 3.3
2. Miscellaneous Exercise 4
3. Miscellaneous Exercise 6
4. Miscellaneous Exercise 8
5. Miscellaneous Exercise 10

The Circuit Value Problem

Definition Boolean circuit

a Boolean circuit is a program that consists of finitely many assignments of form:

$$P_i := 1 \qquad P_i := P_j \wedge P_k \qquad P_i := \neg P_j$$
$$P_i := 0 \qquad P_i := P_j \vee P_k$$

where $j, k < i$ and every P_i is defined at most once

the value of circuit is the value of P_n, where n is maximal

Definition CVP

the circuit value problem (CVP) is defined as:

- given: Boolean circuit (with several inputs)
- question: what is the value of the circuit?

Theorem

The circuit value problem is $\leqslant_m^{\log}$-complete for P

Proof

the proof has two parts

① show $CVP \in \mathrm{P}$

② show CVP is $\leqslant_m^{\log}$-hard for P

Proof ①

easy ∎

Comments

- we already know that the evaluation of a Boolean formula can be performed in LOGSPACE
- while Boolean formulas are labelled trees, Boolean circuits are labelled dags

Proof ②

- let $A \in \mathrm{P}$ and $A = \mathrm{L}(M)$
- M deterministic, single-tape polynomial time-bounded with time-bound n^c
- Q set of states of M
- $\delta \colon Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$
- encode configurations of M in a finite alphabet Δ

Observations

1. runtime is bounded by n^c; memory consumption is bounded by n^c
2. represent computation as $(n^c + 1) \times (n^c + 1)$ matrix with entries in Δ
3. computation can be encoded as local consistency conditions
4. encode these as Boolean circuit

Construction

variables:

$$P^{a}_{ij} \quad (0 \leqslant i,j \leqslant n^c, a \in \Gamma) \qquad\qquad Q^{q}_{ij} \quad (0 \leqslant i,j \leqslant n^c, q \in Q)$$

Local Consistency Conditions

- for $1 \leqslant i \leqslant n^c$, $0 \leqslant j \leqslant n^c$, $b \in \Gamma$:

$$P^{b}_{ij} := \bigvee_{\delta(p,a)=(q,b,d)} (Q^{p}_{i-1,j} \wedge P^{b}_{i-1,j}) \vee$$
$$\vee\ (P^{b}_{i-1,j} \wedge \bigwedge_{p \in Q} \neg Q^{p}_{i-1,j})$$

- for $1 \leqslant i \leqslant n^c$, $1 \leqslant j \leqslant n^c - 1$, $q \in Q$:

$$Q^{q}_{ij} := \bigvee_{\delta(p,a)=(q,b,R)} (Q^{p}_{i-1,j-1} \wedge P^{a}_{i-1,j-1}) \vee$$
$$\vee \bigvee_{\delta(p,a)=(q,b,L)} (Q^{p}_{i-1,j+1} \wedge P^{a}_{i-1,j+1})$$

Local Consistency Conditions (cont'd)

- for $j = 0$:

$$Q^{q}_{i0} := \bigvee_{\delta(p,a)=(q,b,L)} (Q^{p}_{i-1,1} \wedge P^{a}_{i-1,1})$$

- for $j = n^c$:

$$Q^{q}_{in^c} := \bigvee_{\delta(p,a)=(q,b,R)} (Q^{p}_{i-1,n^c-1} \wedge P^{a}_{i-1,n^c-1})$$

- encoding of start configuration on $x = a_1 \dots a_n$:

$$P^{\vdash}_{0,0} := 1 \qquad\qquad P^{a_j}_{0,j} := 1 \ (1 \leqslant j \leqslant n)$$
$$P^{\sqcup}_{0,j} := 1 \ (n+1 \leqslant j \leqslant n^c) \qquad\qquad Q^{s}_{0,0} := 1$$

- all other variables are set to false

assume head moves to the left before accepting, acceptance of M on x is given by

$$Q^{t}_{n^c,0} \vee Q^{t}_{n^c,1}$$

finally the construction is logspace computable ■

The Cook-Levin Theorem

Theorem

Boolean satisfiability is $\leqslant_m^{\log}$-complete for NP

Observations

- with SAT the input values are not given, but have to be found
- CVP is defined in terms of circuits
- SAT is defined in terms of formulas

Proof

additional assignments: $P_i := ?$

rest on blackboard ∎