Algorithm Theory

Georg Moser Mircea Dan Hernest

Institute of Computer Science @ UIBK

Summer 2007

Reductions

Definition — logspace-reductions

L_1 is reducible to L_2 if

1. exists a function R from strings to strings
2. computable by a deterministic TM in space $\mathcal{O}(\log n)$ such that
3. for all x:

$$x \in L_1 \quad \text{iff} \quad R(x) \in L_2 \ .$$

- ➡ REACHABILITY reduces to CIRCUIT VALUE
- ➡ CIRCUIT SAT reduces to SAT
- ➡ CIRCUIT VALUE reduces to CIRCUIT SAT

CIRCUIT VALUE is a special case of CIRCUIT SAT

- ➡ identity as reduction suffices

Theorem — composition of reductions

- R_1 be a reduction from language L_1 to L_2
- R_2 be a reduction from language L_2 to L_3

Then: $R_1 \circ R_2$ is a reduction from language L_1 to L_3

Proof

- M_{R_1} computes R_1; M_{R_2} computes R_2

define $R(x) = R_2(R_1(x))$:

1. start M_{R_2} on x
   if M_{R_2} moves the input cursor
   freeze M_{R_2} and store the cursor position i
2. start M_{R_1} on x on a separate set of strings
3. output of M_{R_1} is written on a work tape
   we only compute the symbol referenced by i
4. resume M_{R_2} $\square$

Completeness

Definition — completeness

$\mathcal{C}$ a complexity class, L is $\mathcal{C}$-complete if

1. $\mathrm{L} \in \mathcal{C}$
2. any language $\mathrm{L}' \in \mathcal{C}$ is (logspace) reducible to L

Example

for the language

$$\mathsf{H}_f = \{\mathsf{M}; x \mid \mathsf{M} \text{ accepts input } x \text{ after at most } f(|x|) \text{ steps}\}$$

1. any $\mathrm{L} \in \mathbf{TIME}(f(n))$ reduces to H_f
2. but $\mathsf{H}_f \notin \mathbf{TIME}(f(n))$

- H_f is not $\mathbf{TIME}(f(n))$-complete

Definition — closure under reductions

a complexity class $\mathcal{C}$ is closed under reductions if, whenever L is reducible to L' and $\mathrm{L}' \in \mathcal{C}$, then $\mathrm{L} \in \mathcal{C}$.

Theorem

P, **NP**, **coNP**, **L**, **NL**, **PSPACE** and **EXP** are closed under reductions.

Theorem

If $\mathcal{C}$, $\mathcal{C}'$ are closed under reductions and $\exists$ L complete for $\mathcal{C}$ and $\mathcal{C}'$, then $\mathcal{C} = \mathcal{C}'$

Proof

we show $\mathcal{C} \subseteq \mathcal{C}'$:

- ➡ let $\mathrm{L}' \in \mathcal{C}$
- ➡ as L is complete for $\mathcal{C}$, L' reduces to L
- ➡ as $\mathrm{L} \in \mathcal{C}'$, $\mathrm{L}' \in \mathcal{C}'$, as $\mathcal{C}'$ closed under reductions

$\mathcal{C}' \subseteq \mathcal{C}$: symmetric □

Definition — computation table

consider a 1-string polynomial-time TM $\mathrm{M} = (K, \Sigma, \delta, s)$ deciding L for fixed x, assume M operates in time-bound $|x|^k$

Represent the computation as a $|x|^k \times |x|^k$ table:

entry (i, j) — contents of position j on the string at time i

Assumptions

1. M halts after at most $|x|^k - 2$ steps (we ignore $|x| = 1$)
2. pad strings if necessary
3. let $\sigma \in \Sigma$, write σ_q, if M reads σ and is in state q
4. cursor starts to the right of $\triangleright$ and never visits $\triangleright$
5. M moves completely to the left before accepting
6. insert identical rows if M halts before $|x|^k - 2$ has expired

Example

$\triangleright$	0_s	1	1	0	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1_{q_0}	1	0	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1	1_{q_0}	0	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1	1	0_{q_0}	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1	1	0	$\sqcup_{q_0}$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1	1	$0_{q_0'}$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1	1_q	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1_q	1	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash_q$	1	1	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$
$\triangleright$	$\vdash$	1_s	1	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$

and so on

$\triangleright$	*yes*	$\vdash$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$	$\sqcup$

Definition

accepting

computation table T is accepting
if $T_{|x|^k-1,1} = \textit{yes}$

Theorem

M as above

M accepts x iff the computation table of M on input x is accepting

Theorem

CIRCUIT VALUE is **P**-complete

Proof

we show that for any $\mathrm{L} \in \mathbf{P}$, there is a (log space) reduction R to CIRCUIT VALUE

- suppose $L = \mathrm{L}(\mathrm{M})$
- M operates within time-bound $|x|^k - 2$
- T denotes $n^k \times n^k$-computation table $\quad n = |x|$

Observations

1. changes in the table from line to the next are local
2. the local changes can be simulated by a circuit C
3. the table is representable by connecting copies of C

Locality

1. let $i = 0$, $j = 0$, or $j = n^k - 1$
   the value of T_{ij} is independent of M and x
2. consider T_{ij}
   equals the symbol under the cursor at position j read at time i

T_{ij} depends only on $T_{i-1,j-1}$, $T_{i-1,j}$, $T_{i-1,j+1}$

$(i-1, j-1)$	$(i-1, j)$	$(i-1, j+1)$
	(i, j)	

Γ denotes all symbols occurring in T

- ➡ encode symbols in Γ in binary
- ➡ define a table S of binary entries

$$S_{ijl} \qquad i \in [0, n^k - 1] \quad j \in [0, n^k - 1] \quad l \in [1, m]$$

$(a_1, \dots, a_m)$	$(b_1, \dots, b_m)$	$(c_1, \dots, c_m)$
	$(d_1, \dots, d_m)$	

- ➡ define m Boolean functions with $3m$ inputs for all i, j

$$S_{ijl} = F_l(S_{i-1,j-1,1}, \dots, S_{i-1,j-1,m}, \dots, S_{i-1,j+1,1}, \dots, S_{i-1,j-1,m})$$

- ➡ Boolean circuit C with $3m$ inputs and m outputs computes

$$F_1 \quad \dots \quad F_m$$

- ➡ given the binary encoding of $T_{i-1,j-1}$, $T_{i-1,j}$, $T_{i-1,j+1}$
  C computes T_{ij}

Facts

1. C depends only on M
2. C has a fixed size independent of x

Definition — Final Construction

construct D_x:

1. $(n^k - 1) \cdot (n^k - 2)$ copies of C
2. the input gates of C_{ij} are identified with the output gates of $C_{i-1,j-1}$, $C_{i-1,j}$, $C_{i-1,j+1}$
3. the input gates of D_x correspond to the first row
4. the single output gate of D_x is the first output of $C_{n^k-1,1}$

Definition — reduction

for every x, set $R(x) = D_x$

construction of the circuit $R(x)$ possible in space $\mathcal{O}(\log n)$ □

Theorem — Cook

SAT is **NP**-complete

Fact

CIRCUIT SAT reduces to SAT

Proof

SAT $\in$ **NP**, the certificate of a polynomial verifier is the assignment

we show that all $\mathrm{L} \in$ **NP** reduce to CIRCUIT SAT:

- ➡ NTM N decides L in time $|x|^k - 2$
- ➡ assume N has at each step two nondeterministic choices
- ➡ the first is called 0, the second 1
- ➡ thus a choice sequence is a string:

$$\vec{c} = (c_0, \dots, c_{n^k-2}) \in \{0,1\}^{n^k-1}$$

Definition

Construction

1. construct a computation table
   replace S_{ijl} by S_{i,j,l,c_i}:

$(a_1,\dots,a_m)$	$(b_1,\dots,b_m)$	$(e_1,\dots,e_m)$	c_i
	$(d_1,\dots,d_m)$		

2. recall that m denotes the bit-length of the symbols in the computation table
3. circuit C has $3m+1$ inputs
4. the extra arguments from $\vec{c}$ become input gates
5. construction of the circuit $R(x)$ possible in space $\mathcal{O}(\log n)$

$\square$