§9 Memory (=*Space*)

KAIST
CS500 M. Ziegler

"Time is unbounded, memory not!"

Examples: a) Integer addition in logarithmic space. decision
b) Integer multiplication in logarithmic space.

$x_{n-1}\ldots x_k\ldots x_1\, x_0 \times y_{n-1}\ldots y_1\, y_0$

$x_{n-1}\ldots x_k\ldots x_1\, x_0$

$x_{n-1}\ldots x_k\ldots x_1\, x_0$

$x_{n-1}\ldots x_k\ldots x_1\, x_0$

$+c_k \le 2n$

$m=2n$

$+$ $x_{n-1}\, x_{n-2}\ldots x_k\ldots x_2\, x_1\, x_0$ / $y_{n-1}\, y_{n-2}\ldots y_k\ldots y_2\, y_1\, y_0$ $\mathrm{bin}(k)$

$= z_n\, z_{n-1}\, z_{n-2}\ldots z_k\ldots z_2\, z_1\, z_0$ $\quad m=n+1$

$n = \ell(x) := \lceil \log_2(x+1) \rceil$
bin.*length* of $x\in\mathbb{N}$

time $T \Rightarrow$ space $S\le T$

space $S \Rightarrow$ time $T\le O(2^S)$

op. $Q(j)$: j-th bit of input, $0\le j<n$

- charge algo. for #bits stored
- aim for *sub*-linear space,
- input *not* counted: read-only
- output: **0/1** decision problem

Graph Reachability/Savitch

KAIST
CS500 M. Ziegler

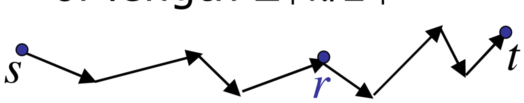
Walter Savitch: Graph reachability in space $O(\log^2 n)$.

Examples: a) Integer addition in logarithmic space.
b) Integer multiplication in logarithmic space.

Given un-/directed graph $G=(V,E)$ and $k\in\mathbb{N}$, $s,t\in V$: $\quad n=|V|$, $m=|E|$
Is there a path from s to t in E of length $\le k$?

Recursively test for all $r\in V$ whether $\quad S(k) \le O(\log n) + S(k/2)$

- there is a path from s to r in E of length $\le \lfloor k/2 \rfloor$
- and one from r to t of length $\le \lceil k/2 \rceil$



space $S \Rightarrow$ time $T\le O(2^S)$

op. $Q(j)$: j-th bit of input, $0\le j<n$

- charge algo. for #bits stored
- aim for *sub*-linear space,
- input *not* counted: read-only
- output: **0/1** decision problem

Matrix Multiplication/Powering

Walter Savitch: Graph reachability in space $O(\log^2 n)$.

Examples: a) Integer addition in logarithmic space.

b) Integer multiplication in logarithmic space.

c) Integer matrix multiplication $\{0,\ldots 2^\ell-1\}^{d\times d} \ni A,B \to A\cdot B$ in space $O(\log \ell+\log d)$

$\Sigma_{v=1..n}\, a_{uv}\cdot b_{vw}$ length≈$2\ell+\log d$

d) Integer matrix powering $\{0,\ldots,2^\ell-1\}^{d\times d} \ni A \to A^k$

Using repeated squaring: space $O(\log(\ell\cdot d\cdot k)\cdot\log(k))$

length ≈ $2^j\cdot\ell + (2^j-1)\cdot\log d$

$j=1..\log k$

$= k\cdot\ell+(k-1)\cdot\log d$

(a) to (c) are optimal!

space $S \Rightarrow$ time $T\leq O(2^S)$

op. $Q(j)$: j-th bit of input, $0\leq j<n$

- charge algo. for #bits stored
- aim for *sub*-linear space,
- input *not* counted: read-only
- output: **0/1** decision problem

Composition and Iteration

Theorem: If $\mathcal{A}$ computes family $f:\{0,1\}^n\to\{0,1\}^{m(n)}$

in space $S(n)$ $\geq \log(m(n)) + \log(n)$

and $\mathcal{B}$ computes family $g:\{0,1\}^m\to\{0,1\}^{k(m)}$

in space $R(m)$ $\geq \log(k(m)) + \log(m)$,

then $\mathcal{B}\circ\mathcal{A}$ computes family $g\circ f:\{0,1\}^n\to\{0,1\}^{k(m(n))}$

in space $O(R(m(n))+S(n))$

Corollary:

If $f:\{0,1\}^n\to\{0,1\}^{\ell+n}$ computable in space $S(n)\geq\log(n)$,

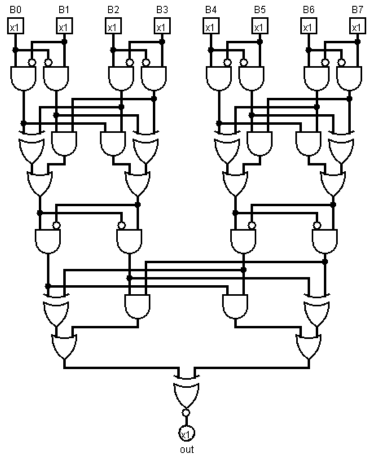
its k-fold iteration $f^k:\{0,1\}^n\to\{0,1\}^{k\ell+n}$ is computable

in space $O(k\cdot S(k\ell+n))$.

space $S \Rightarrow$ time $T\leq O(2^S)$

Main examples $S(n) = O(\log n)$

Parallel Computing: Circuits



Boolean circuit:

=partial order

- directed acyclic graph
- unary ¬ and <u>binary</u> ∧, ∨
- with n inputs
- of depth d size $\leq d \cdot w$
- and width w $\leq 2^d$

=maximal anti-chain

Theorem: Any Boolean circuit of depth d and width w can be simulated in space $O(d)$ and in time $O(d \cdot w)$.

A computation in time T and space S can be simulated by a Boolean circuit of depth $O(S \cdot \log T)$ and width $O(4^S)$

space $S \Rightarrow$ time $T \leq O(2^S)$

Main examples $S(n) = O(\log n)$