Space Complexity — Savitch's Theorem

Space Complexity of TMs

The **space complexity** of a deterministic Turing machine M

$\cdots$ for each n, it is the maximum number of tape cells that M's head touches in processing any input of length n

The **space complexity** of a nondeterministic Turing machine M

$\cdots$ take the maximum for all computation paths of M on all inputs of length n

For any $f : \mathcal{N} \rightarrow \mathcal{N}$, a Turing machine M is $f(n)$ **space** if M's space complexity is f.

Note: Time bounded Turing machines always halt, but space bounded Turing machines may or may not halt.

(By a theorem of Sipser, if f is *space-constructible* – see p. 336 Sipser – we can convert to an f-space TM that always halts, i.e., a decider.)

Whether the input can be modified or not

For functions f such that $f(n) < n$ for some n, we treat the space complexity differently:

- the Turing machine has more than one tape,
- the input is given with **markers indicating two ends** and **will never be modified,**
- only the work tapes are considered in measuring the space complexity of a computation.

Remark: In some versions, intended to deal with space complexities smaller than *log* n, markers are used to indicate the two ends of the allowable space of $f(n)$ cells on each work tape. But $f(n)$ cannot be an arbitrary, perhaps non-computable function, or else we could prove undecidable problems to be $O(1)$-space decidable!

Space Complexity Classes

Definition. *Let $f : \mathcal{N} \rightarrow \mathcal{N}$ be a function. Then* $\mathrm{SPACE}(f(n)) = \{L \mid L$ *is decided by an $O(f(n))$ space deterministic Turing machine* $\}$*, and.* $\mathrm{NSPACE}(f(n)) = \{L \mid L$ *is decided by an $O(f(n))$ space nondeterministic Turing machine* $\}$*.*

Example: $SAT \in \mathrm{SPACE}(n)$**.**
Consider a 2-tape TM that, on input a formula ϕ of n variables, **tries all assignments to ϕ,** each viewed as an n bit binary number, and accepts if and only if one assignment that satisfies ϕ has been found.

Such a machine correctly decides SAT and $O(n)$ space.

Space Complexity May Depend on Encoding Schemes

Example: $CLIQUE \in \mathrm{SPACE}(\sqrt{n})$
Suppose that the adjacency matrix is used for representing a graph.

Consider a 2-tape TM that, on input of a graph $G = (V, E)$ of some m nodes and a number $k \geq 1$, **tries all possible subsets of V,** viewed as an m-bit binary number, and accepts if and only if one subset is of size k and is a clique in G.

Let the encoding length be n. Then $n \geq m^2$ and the space requirement is $O(m)$. So, $CLIQUE \in \mathrm{SPACE}(\sqrt{n})$.

Savitch's Theorem

Theorem. *Suppose $f(n) \geq C \log n$ for some constant $C > 0$. Then $\mathrm{NSPACE}(f(n)) \subseteq \mathrm{SPACE}(f(n)^2)$.*

Proof Let $f(n) \geq \log n$, $L \in \mathrm{NSPACE}(f(n))$ via a 2-tape NTM M with an unmodifiable input tape and a work tape. Let $x, |x| = n$, be an input. Since a marker is provided at each end of each tape, we can assume that M erases the work tape cells and moves the heads to the leftmost positions before halting so that **there is a unique accepting configuration.**

Encoding Configurations

Each configuration of M on x is a combination of the **state**, the **input head position**, the **work tape head position**, and the **work tape contents**, requiring $O(1)$ symbols, $O(\log n)$ symbols, $O(\log f(n))$ symbols, and $f(n)$ symbols, respectively.

Since $f(n) \geq C \log n$, the total number of symbols needed is at most $kf(n)$ for some fixed k.

There are at most $2^{lf(n)}$ configurations for some fixed l.

Reachability

For configurations c, c' and an integer $t \geq 0$, define

$CANYIELD(c, c', t) = 1$ if c yields c' within 2^t steps and 0 otherwise.

Let c_0 be the initial configuration and c_{accept} be the unique accepting configuration. Then $x \in L$ if and only if $CANYIELD(c_0, c_{accept}, lf(n)) = 1$.

For each c, c', $CANYIELD(c, c', 0) = 1$ if and only if $c \Rightarrow c'$, so whether $CANYIELD(c, c', 0) = 1$ can be tested by simulating all possible single step moves of the TM from configuration c.

For each c, c', and $t > 0$, $CANYIELD(c, c', t) = 1$ if and only if there is some d such that $CANYIELD(c, d, t-1) = CANYIELD(d, c', t-1) = 1$.

Deterministic Recursive Reachability Testing

Call $CANYIELD(c_0, c_{accept}, lf(n))$ and accept if and only if 1 is returned.

$CANYIELD(c, c', t)$
if $t = 0$ **then** simulate all possible one-step moves of M
  from c, and return 1 if c' is generated
else
  for each configuration d
    if $CANYIELD(c, d, t-1) = CANYIELD(d, c', t-1) = 1$
      then return 1.
  return 0.

The recursion depth is t, so the space requirement is $O(kf(n) \cdot lf(n)) = O(f(n)^2)$. ∎