Umans Complexity Theory Lectures

Lecture 1d: Linear Speedup

Complexity Classes

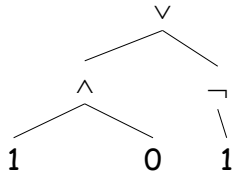
- Back to complexity classes:
  - **TIME(T(n))** = languages decidable by a multi-tape TM in at most T(n) steps, where n is the input length, and T :**N** → **N**
  - **SPACE(S(n))** = languages decidable by a multi-tape TM that touches at most S(n) cells of its work tapes, where n is the input length, and S :**N** → **N**

Note: $\mathbf{P} = \cup_{k \geq 1} \mathbf{TIME}(n^k)$

Time and Space

A motivating question:

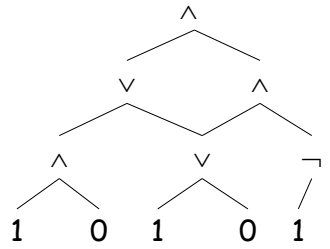
- Boolean formula with n nodes
- evaluate using O(log n) space?



- depth-first traversal requires storing intermediate values
- idea: short-circuit ANDs and ORs when possible

Time and Space

- Can we evaluate an n node Boolean **circuit** using O(log n) space?



Time and Space

- Recall:
  - **TIME(T(n))**, **SPACE(S(n))**
- Questions:
  - how are these classes related to each other?
  - how do we define **robust** time and space classes?
  - what problems are contained in these classes? complete for these classes?

Outline

- Why big-oh? Linear Speedup Theorem
- Deterministic Hierarchy Theorems
- Robust Time and Space Classes
- Relationships between classes
- Some complete problems

Linear Speedup

Theorem: Suppose TM M decides language L in time T(n). Then for any $\varepsilon > 0$, there exists TM M' that decides L in time

$$\varepsilon\, T(n) + n + 2.$$

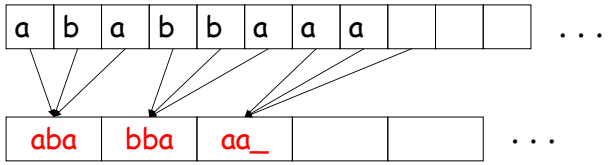
- Proof:
  - simple idea: increase "word length"
  - M' will have
    - one more tape than M
    - m-tuples of symbols of M

$$\Sigma_{new} = \Sigma_{old} \cup \Sigma_{old}^{m}$$

    - many more states

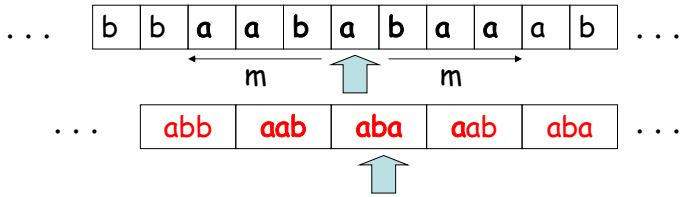
Linear Speedup

- part 1: compress input onto fresh tape



Linear Speedup

- part 2: simulate M, m steps at a time



  - 4 (L,R,R,L) steps to read relevant symbols, "remember" in state
  - 2 (L,R or R,L) to make M's changes

Linear Speedup

- accounting:
  set m = 6/ε
  - part 1 (copying): n + 2 steps
  - part 2 (simulation): 6 (T(n)/m)= 6 (T(n)/(6/ε))= εT(n)
  - Total time: εT(n) + n + 2

Theorem: Suppose TM M decides language L in space S(n). Then for any ε > 0, there exists TM M' that decides L in space ε S(n) + 2.

- Proof: same.

Time and Space

- Moral: big-oh notation **necessary** given our model of computation
  - Recall: $f(n) = O(g(n))$ if there exists c such that $f(n) \leq c\, g(n)$ for all sufficiently large n.
  - TM model incapable of making distinctions between time and space usage that differs by a constant.
- In general: interested in course distinctions not affected by model
  - e.g. simulation of k-string TM running in time T(n) by single-string TM running in time $O(T(n)^2)$