Computability theory vs. complexity theory I

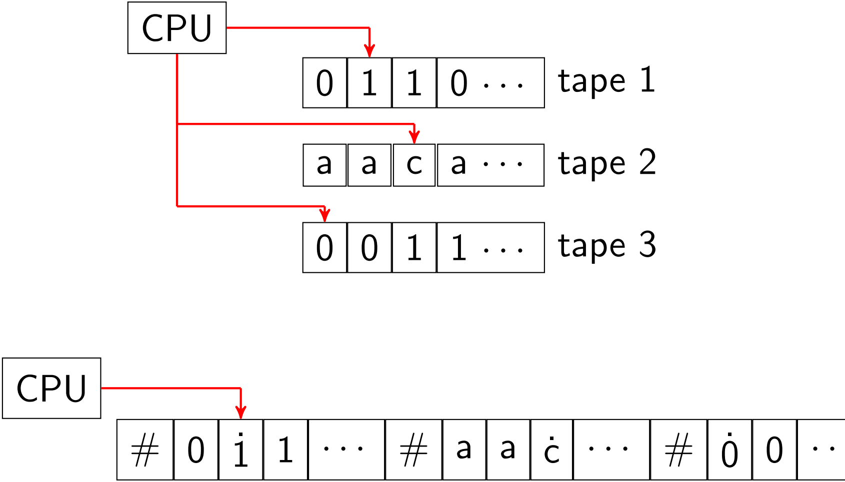
- In Chapter 3 we showed that various TMs are equivalent
- For example, single-tape and multi-tape TMs are equivalent
- However, their "time complexities" are different

Complexity of Multi-tape TM I

- Theorem 7.8

  Let $t(n) \geq n$. For a $t(n)$ multi-tape TM
  $\Rightarrow \exists$ equivalent $O(t(n)^2)$ single-tape TM
- Idea for the proof: similar to how we proved their equivalence
- We will show that simulating each step of a multi-tape TM takes $O(t(n))$ on a single-tape TM
- Let k be the number of tapes
- How did we simulate a multi-tape TM?

Complexity of Multi-tape TM II



Complexity of Multi-tape TM III

- To simulate each step of multi-tape TM, we scan to know where heads point to and do the update
- However, we may have to right shift the tape
- So we need to know the tape length. It is

$$k \times O(t(n)) = O(t(n))$$

- Note that each tape of multi-tape TM has $O(t(n))$ length. Why?
- A $t(n)$ multi-tape TM generates $O(t(n))$ contents in $O(t(n))$ time

Complexity of Multi-tape TM IV

- Thus the cost of simulating each step of multi-tape TM on a single-tape TM is $O(t(n))$
- There are $O(t(n))$ multi-tape TM steps, so the total cost is

$$O(t(n)) \times O(t(n)) = O(t(n)^2)$$

Complexity of Nondeterministic TM I

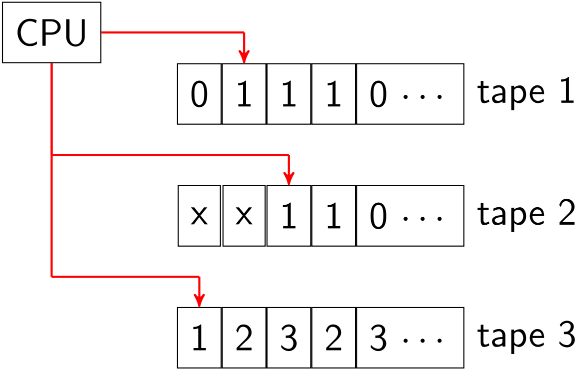
- Remember NTM is a decider if all branches halt on all inputs
- Definition of NTM time complexity $t(n)$:

  maximum # of steps the machine uses for any path from root to a leaf
- Theorem 7.11

  Let $t(n) \geq n$. For a $t(n)$ NTM (single tape) $\Rightarrow \exists$ a $2^{O(t(n))}$ TM (single tape)
- Assume b is the maximal number of branches at each layer

Complexity of Nondeterministic TM II

- Recall our way of doing the simulation is by the following three-tape TM



Complexity of Nondeterministic TM III

- We use a breadth-first way for the simulation
- That is, after one layer is finished, we do the next
- Tape 3: all possible paths so far
- Total number of nodes in the tree:

$$O(b^{t(n)})$$

- Tape 2: run the original input w from root to one node in the tree
- Cost of running from root to one node in tape 2: $O(t(n))$

Complexity of Nondeterministic TM IV

- Update of tape 3: $O(b^{(t(n))})$
- Total cost $\leq O(b^{t(n)})$ for each node
- Total time:

$$
\begin{aligned}
& \# \text{ nodes } \times \text{ cost per node} \\
= & O(b^{t(n)}) \times O(b^{t(n)}) \\
= & O((b^2)^{t(n)}) = 2^{O(t(n))}
\end{aligned}
$$

- Note that in the above equality we used

$$
b^{t(n)} \times b^{t(n)} = b^{2t(n)} = 2^{\log_2 b^{2t(n)}} = 2^{(\log_2 b)(2t(n))}
$$

Complexity of Nondeterministic TM V

- This is by a 3-tape TM
- To use a single-tape TM to simulate a 3-tape one, we need

  $$(2^{O(t(n))})^2 = 2^{O(t(n))}$$

  cost because

  $$(2^{O(t(n))})^2 \leq (2^{ct(n)})^2 = 2^{2ct(n)} = 2^{O(t(n))}$$